

University of Dundee

MASTER OF PHILOSOPHY

A Bliss Writing System

Nandadasa, Palatuwa Gamage Rajitha Hasith

*Award date:*  
2021

[Link to publication](#)

#### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

#### **Take down policy**

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.



# A Bliss Writing System

Palatuwa Gamage Rajitha Hasith Nandadasa

*This thesis is submitted in partial fulfilment for the degree of  
Master of Philosophy at the University of Dundee*

University of Dundee

October 2021

# Acknowledgements

I would like to thank my supervisor Professor Annalu Waller for her continuous guidance, advice and support throughout the last three years. Thank you for believing in me even in times that I was not. I would also like to thank Kathleen Cummins for her continuous support and encouragement during my research work. It has not been an easy ride and I could not have done this without my AAC research Group colleagues, Chris Norrie, Ladan Safiee, Rolf Black, Michael Everson and Elaine Scougal, thank you for the support and encouragement that you have given me for the last three years. A special thanks to Michael for providing me many different versions of his draft Unicode font for Blissymbols throughout my research; obviously such a font is vital for the use of Blissymbols as a writing system .

I would like to thank the participants for their participation in my research and the feedback that you have given me. Your contribution is very much appreciated, and I would not have been able to complete this work without you. I would like to thank Kathleen again for helping me with organising the participants and keeping me in contact with the participants.

Finally, I would like to thank my family for everything they have done for me for the past three years. I cannot imagine doing this without your support, encouragement, and love. I am very fortunate to have an Uncle and Aunt who support me in my whole life in so many different ways. Specially the last ten years for supporting me financially, providing me food and lodging but most importantly for the love and guidance to be a better person every day. Thank you for being my home away parents. My cousins (obviously brothers and sisters) who live in Scotland and in Sri Lanka, I am very much indebted to you for keeping me sane throughout my studies. Specially my two sisters who live with me, thank you for the love and support and very much needed teasing. And again finally, my parents, even though I am away from home you have been with me every single day by giving me love and support. It must have been very hard for you to send me to a country that is far from you. Thank you for raising me to be a good person and surrounding me with a very much loving family.

# **Declarations**

## **Candidate's Declaration**

I, Rajitha Hasith Nandadasa, hereby declare that I am the author of this thesis; that all references cited have been consulted by me; that the work of which this thesis is a record has been done by myself; and that this thesis has not been previously accepted for a higher degree.

P. G. Rajitha Hasith Nandadasa

## **Supervisor's Declaration**

I, Professor Annalu Waller, hereby declare that I am the supervisor of the candidate, and that the conditions of the relevant Ordinance and Regulations have been fulfilled.

Professor Annalu Waller

# Abstract

Being able to communicate in the natural environment and sufficiently meet daily communication needs is the essence of communication. The majority of the population in the world does not think about the mechanics required to communicate effectively. However, there is a significant number of people who experience difficulties in communication due to speech and/or cognitive impairments. Alternative methods of communication can be used to help people with disabilities to maximise the competence of communication. These alternative methods are referred to as Augmentative and Alternative Communication (AAC).

Blissymbolics (typically shortened to “Bliss”) is an ideographic written language that was developed by Charles K. Bliss for international communication in 1949, but currently Bliss is used by pre-literate nonspeaking children and adults as a means to communicate. In the past and present, AAC access methods that use Blissymbolics have been mainly developed by the teacher or the therapist rather than the Bliss user. Bliss users typically access Bliss from a predefined chart, that is configured by the teacher/therapist to form sentences, which would then be voiced by a listener or output as speech using a high-tech AAC device. Bliss has mainly been used as a symbol set rather than a written language. Bliss users have never been given access to the full 1400+ Bliss-character set which would allow the Bliss users to write in Blissymbolics rather than access Bliss as a symbol set.

To address this the researcher developed a set of three input methods for Bliss-characters which gives the Bliss user access to the full Bliss-character set:

1. An input method based on the Bliss-alphabet
2. An input method based on glyph elements (strokes) used in Bliss-characters
3. A hybrid input method based on both of these

The Bliss-alphabet-based input method allows a Bliss user to access Bliss-characters by how the characters are categorised by the Bliss-alphabet shape. This allows the

Bliss user to input a Bliss-alphabet shape and access Bliss-characters that belong to that letter of the Bliss-alphabet.

The stroke-based input method for Bliss allows a user to access Bliss-characters using the basic shapes (similar to strokes in Chinese writing) that have been used to form these Bliss-characters. After analysing how each Bliss-character is formed using these shapes, the researcher implemented an input method which allows a Bliss user to type multiple Bliss-alphabet shapes and access all the Bliss-characters that are formed with the typed Bliss-alphabet shapes.

The hybrid-based input method is a combination of both the Bliss alphabet-based input method and the Bliss stroke-based input method. In the hybrid-based input method, Bliss-characters are accessed using both Bliss alphabet-based and stroke-based input methods; when user type the first Bliss-alphabet shape, the Bliss-characters are retrieved using the Bliss alphabet-based input method, after which Bliss-characters are retrieved using the Bliss stroke-based input method.

The researcher ran two user studies with participants with various physical and speech impairments who use different AAC systems and access methods. The first study was conducted with seven participants in which they evaluated the Bliss writing system. The system log data from study 1 led into the implementation of the Bliss hybrid-based input method.

Study 2 was conducted with one participant who has some Bliss knowledge. The purpose of the study was to evaluate the Bliss hybrid-based input method and to compare the three Bliss-character input methods in terms of the speed of writing using each Bliss-character input method.

The results from study 2 show that the Bliss hybrid-based input method is faster than the Bliss alphabet-based and stroke-based input methods in terms of writing in Blissymbolics.

# Table of Contents

<b>Acknowledgements.....</b>	<b><i>i</i></b>
<b>Declarations .....</b>	<b><i>ii</i></b>
<b>Abstract.....</b>	<b><i>iii</i></b>
<b>List of Tables.....</b>	<b><i>ix</i></b>
<b>List of Figures.....</b>	<b><i>xi</i></b>
<b>1 Introduction .....</b>	<b>1</b>
1.1 Structure of the Thesis .....	2
<b>2 Background.....</b>	<b>4</b>
2.1 Augmentative and Alternative Communication.....	4
2.2 Access to AAC Systems by Users with Physical Impairments .....	6
2.2.1 Pointing Devices .....	6
2.2.2 Switch Scanning .....	6
2.2.3 Eye-Gaze .....	7
2.3 Natural Language Processing in AAC .....	7
2.4 Blissymbolics .....	9
2.4.1 Background.....	9
2.4.2 Blissymbolics in AAC .....	11
2.5 Fundamental Rules of Blissymbolics .....	12
2.5.1 Bliss Alphabet .....	13
2.5.2 Bliss-Characters .....	14
2.5.3 Bliss-Words .....	14
2.5.4 Indicators .....	15

2.5.5	Bliss Sentences .....	17
<b>2.6</b>	<b>Blissymbolics and Technology.....</b>	<b>17</b>
2.6.1	Bliss-Talk .....	18
2.6.2	BlissOnline.se.....	19
2.6.3	Bliss-Dasher .....	19
2.6.4	MisterBLISS .....	20
2.6.5	Simplipedia .....	21
2.6.6	Bliss OpenOffice Plug-in .....	21
<b>2.7</b>	<b>Bliss as a Character Set .....</b>	<b>22</b>
2.7.1	Accessing Bliss-characters .....	23
<b>3</b>	<b><i>Initial Keyboard Design.....</i></b>	<b>27</b>
<b>3.1</b>	<b>The Development of the Bliss Writer .....</b>	<b>27</b>
3.1.1	Bliss-Alphabet Mapping onto the QWERTY Keyboard .....	30
<b>3.2</b>	<b>Bliss Writer - System Design .....</b>	<b>33</b>
3.2.1	User Stories.....	33
3.2.2	Bliss-character Input methods .....	34
<b>4</b>	<b><i>Study 1.....</i></b>	<b>38</b>
<b>4.1</b>	<b>Participants.....</b>	<b>38</b>
<b>4.2</b>	<b>Method .....</b>	<b>39</b>
<b>4.3</b>	<b>The Bliss Writer Study 1 Results.....</b>	<b>41</b>
4.3.1	Participant 1 – P1.....	42
4.3.2	Participant 2 – P2.....	44
4.3.3	Participant 3 – P3.....	44
4.3.4	Participant 4 – P4.....	46
4.3.5	Participant 5 – P5.....	47
4.3.6	Participant 6 – P6.....	47



4.3.7	Participant 7 – P7 .....	48
4.4	Study 1 Analysis and Discussion .....	50
<b>5</b>	<b><i>The Bliss Stroke-based Large Character Set Problem.....</i></b>	<b>53</b>
5.1	Bliss Hybrid-based Input Method.....	55
<b>6</b>	<b><i>Study 2.....</i></b>	<b>56</b>
6.1	Participants.....	56
6.2	Method .....	56
6.3	The Bliss Writer Study 2 Results.....	58
6.3.1	Participant 1 - P1 .....	58
6.4	Study 2 Analysis and Discussion.....	59
<b>7</b>	<b><i>Discussion .....</i></b>	<b>63</b>
7.1	Providing Keyboard Access to the Full Bliss-Character Set.....	64
7.2	The Experience of Disabled Users using the Bliss Writer .....	65
7.3	The Large Character Set Problem .....	66
7.4	Limitations of the Research .....	67
7.5	Future Work .....	68
7.5.1	Implementing further Bliss-Character input methods .....	69
7.5.2	Natural Language Processing in Blissymbolics .....	70
7.5.3	Further Development on Bliss Writer System.....	70
<b>8</b>	<b><i>Conclusion.....</i></b>	<b>72</b>
<b>9</b>	<b><i>References.....</i></b>	<b>77</b>
	<b><i>Appendix A. ....</i></b>	<b>82</b>
	<b><i>Appendix B. Ethics Approval From .....</i></b>	<b>83</b>
	<b><i>Appendix C. Study 1 Questionnaire.....</i></b>	<b>84</b>

<b><i>Appendix D. Bliss-Writer Development</i></b>	<b>86</b>
<b>Tech Stack</b>	<b>86</b>
Meteor Web Framework	86
MongoDB	86
<b>Implementation</b>	<b>87</b>
Application Structure	87
Database Structure	90

# List of Tables

Table 1: Bliss- Alphabet Letters .....	13
Table 2: Examples of Bliss-characters .....	14
Table 3: Examples of Bliss-words .....	15
Table 4: Nominal indicators .....	15
Table 5: Verbal indicators .....	16
Table 6: Adjectival and Adverbial Indicators .....	16
Table 7: Examples of Bliss Indicators use .....	16
Table 8: Some Bliss-characters and their associated Bliss-alphabet shapes .....	26
Table 9: Few Bliss-characters from Bliss Stroke-based Input method Analysis .....	36
Table 10: Study Participants, Access Method and their Bliss knowledge .....	38
Table 11: Study Participant’s session details .....	42
Table 12: Participant 1 Bliss alphabet-based Keyboard Sentences .....	43
Table 13: Participant 2 Bliss alphabet-based Keyboard Sentences .....	44
Table 14: Participant 3 Bliss alphabet-based Keyboard Sentences .....	45
Table 15: Participant 4 Bliss alphabet-based Keyboard Sentences .....	46
Table 16: Participant 6 Bliss alphabet-based Keyboard Sentences .....	48
Table 17: Participant 6 Bliss Stroke-based Keyboard Sentences .....	48
Table 18: Participant 7 Bliss alphabet-based Keyboard Sentences .....	49
Table 19: Participant 7 Bliss Stroke-based Keyboard Sentences .....	50
Table 20: Participants attempts on writing Bliss-word “lake” .....	51
Table 21: Number of Bliss-Characters Displayed to the user with one Bliss-Alphabet Shape .....	54
Table 22: Study Participants, Access Method and their Bliss knowledge .....	56

Table 23: Participant's Copied Bliss Sentences .....	59
Table 24: Bliss-Character Input methods Comparison .....	59
Table 25: Variations of Bliss-Alphabet Shapes .....	69
Table 26: MongoDB and Relational Database Comparison .....	87

# List of Figures

Figure 1: A Creature Who Drinks Blood at Night .....	11
Figure 2: Wubi 86 Keyboard Layout (from Wikipedia) .....	24
Figure 3: Cangjie Keyboard Layout (from Wikipedia).....	25
Figure 4: Blissymbol drawing stencil (Bliss, 1965).....	28
Figure 5: Blissymbol Stencil Template (Everson, 2020) .....	28
Figure 6: First set of keyboard keys, which we find on any typewriter (Bliss, 1965).....	29
Figure 7: Second set of keyboard keys represent the strokes for composition of the Blissymbols (Bliss, 1965) .....	29
Figure 8: Dot printed using Semantographic Typewriter with a Steering stick (Bliss, 1965) .....	30
Figure 9: First Iteration Bliss Keyboard Mapped to QWERTY Layout (2018) (Everson, 2020) .....	31
Figure 10: Updated Bliss Keyboard Layout (Everson, 2020).....	31
Figure 11: Bliss Onscreen Keyboard .....	32
Figure 12: Bliss Onscreen Keyboard (Shift-pressed).....	32
Figure 13: English Alphanumeric keyboard in QWERTY layout .....	33
Figure 14: Bliss Writer – Alphabet-based Keyboard .....	35
Figure 15: Bliss Writer – Stroke-based Keyboard .....	37
Figure 16: Bliss Writer – Hybrid Keyboard.....	55
Figure 17: Example Project Structure .....	88
Figure 18: Bliss-Characters DB Collection Schema.....	90

# 1 Introduction

Communication is a critical function for a human being. Functional skills in communication are the skills that are required to initiate and maintain daily interactions within the natural environment (Light, 1989). This could be as simple as asking directions from a stranger or telling a story to a friend, or as complicated as explaining how a railway engine works. Siegel (2002) argues that without communication an individual cannot become an effective and informed citizen in our society. Communication is intuitive, and most people do not think about the mechanics required to communicate effectively. But there is a significant number of people who experience difficulties in communication due to speech and/or cognitive impairments. Augmentative and Alternative Communication (AAC) can be used to help people with these impairments. In the United Kingdom over 1.4 million children and young people have speech, language, and communication needs (SLCN). (Scottish Government, 2012).

Augmentative and Alternative Communication is an umbrella term that is used to describe different methods of communication that can “add-on” to speech and to get around difficulties with ordinary speech (Communication Matters, n.d.). A variety of symbol sets such as Picture Communication Symbols (PCS) and Widgeit (Rebus Symbols) are available and are implemented on a variety of platforms, ranging from communication boards and books to high-tech electronic AAC systems such as Dynavox devices and Widgeit software (Communication Matters, 2017). Blissymbolics is an ideographic written language whose characters have semantics but no phonology (unlike Chinese, which is logographic, where the characters typically have both a semantic and a phonetic component). In 1947, when Charles Bliss published the book *Semantography* (Blissymbolics) his intention was to use Bliss as a universal language, however Blissymbolics is primarily used for AAC. Blissymbol boards and books provide access to static Bliss-words. Some electronic AAC systems such as Blissymbol Minspeak Words (Ulrika, et al., 2009) allow users to use Blissymbols to access pre-stored words and phrases which can be spoken using speech synthesis. To date, Blissymbolics has been treated as a symbol set in terms of aided AAC.

This research aims to investigate ways in which disabled users can access Blissymbolics as a written language, not as a symbol set. This thesis will describe Blissymbolics and how this writing system differs from AAC symbol sets. The focus of the current research is to enable Blissymbolics users to “write” by designing an efficient and usable interface. This will necessarily involve processing Blissymbolics as a natural language and demonstrating how users can explore the generative capabilities of a semantic written language.

The researcher hypothesises the following in this research project:

***It is possible to design and implement a writing system that allows disabled Bliss users to generate Bliss-words by writing using Blissymbolics.***

To test this hypothesis, the following research questions will be addressed in this research.

- RQ1. Is it possible to design a keyboard input method for the Bliss-character set?*
- RQ2. What is the experience of people with physical impairments when accessing Bliss-characters using different access methods?*
- RQ3. How do different keyboard input methods impact on the speed of typing Bliss?*

In order to answer these research questions, the researcher developed the Bliss Writer system with three different Bliss-character input methods. To test the Bliss writer system and evaluate the Bliss-character input methods two user studies were conducted.

## **1.1 Structure of the Thesis**

This thesis’ contribution to science is to give access to Blissymbolics as a written language expressed as text, rather than as a symbol set expressed by pictures. This was achieved by developing three different Bliss-character input methods, one based on

the Bliss-alphabet, one based on Bliss-strokes, and a hybrid-based input method, a combination of alphabet-based and stroke-based input methods.

The background section (chapter 2) provides an introduction to Augmentative and Alternative Communication (AAC), discusses different AAC categories and the various systems which are in those categories. This chapter also explores different access methods in AAC in terms of direct and indirect access methods. Section 2.4 gives an introduction to Blissymbolics with a background on how Bliss was invented and how Bliss is used in AAC. Section 2.5 describes the fundamentals of Bliss with some examples. This section will give some insight into Blissymbolics and how it operates as a written language. After that section 2.6 investigates existing technologies that have been developed to access Bliss and finally explores Blissymbolics as a character set along with the challenges that there are with character input methods for large character sets such as Bliss. Chapter 3 describes the initial Bliss keyboard design and the implementation of the Bliss Writer system along with Bliss-character input methods. Chapter 4 describes study 1 that was conducted with seven participants. This chapter describes the participants, methodology, results and analysis of the study. Chapter 5 introduces a new Bliss-character input method called hybrid-based input method. This chapter describes how the Bliss hybrid-based input method was implemented. Chapter 6 is about study 2 that was conducted after implementing the Bliss hybrid-based input method. The chapter describe the methodology, results and discusses the analysis of the study results. Chapter 7 is an overall discussion of the research on how the research questions were addressed. This also includes the limitations of the research and, finally, suggests future work. In the final chapter the researcher summarises the thesis and conclusion of the overall research.



## **2 Background**

This chapter gives an introduction to Augmentative and Alternative Communication (AAC) and different types of AAC access methods. The chapter also dives in to background of Blissymbolics, rules of how Blissymbolics is used as a written language, technologies developed for Blissymbolics and look in to Bliss as a character set.

### **2.1 Augmentative and Alternative Communication**

This section gives an introduction to Augmentative and Alternative Communication and describes different types of AAC systems.

Augmentative and Alternative Communication (AAC) can involve unaided and aided communication methods. Unaided AAC includes sign language, gestures, and facial expressions, while aided AAC systems can range from communication boards to electronic AAC systems that are capable of word prediction and speech generation. Aided AAC systems can be divided into low-tech and high-tech AAC systems (Beukelman & Mirenda, 2014)

- Low-Tech AAC systems
  - This category includes methods that do not need any battery or power to function. Examples are pen and paper, letter boards, word boards, symbol carts, communication charts, etc.
- High-Tech AAC systems
  - This category ranges from a simple button that speaks a pre-recorded phrase or words when pressed to very advanced systems that are capable of text to speech and language prediction systems.

High-tech AAC systems have evolved over time and have become more advanced. There is a wide range of high-tech AAC systems. These include systems using natural

language processing such as word prediction. Smartphones and tablets have also offered many potential benefits to AAC users since their release. McNaughton & Light (2013) state that such benefits include increased awareness and social acceptance of AAC in the mainstream, greater consumer empowerment in accessing AAC solutions, increased adoption of AAC technologies, greater functionality and interconnectivity, and increased opportunities for a wide range of programmers (family members, students, clinicians, and mainstream programmers) to develop apps for AAC users. McNaughton and Light also address both the challenges and the benefits of these mobile technologies (2013):

- keeping the focus on communication, not just technology;
- developing innovative approaches to AAC assessment and intervention;
- ensuring ease of access to AAC for all individuals with complex communication needs;
- maximizing AAC solutions to support a wide variety of communication functions;

AAC systems provide users with access to meaning in the form of signs, images, symbols, or the written word. Several AAC symbol sets have been developed which represent a range of vocabulary items (that is, words and phrases). Symbols that represent concrete meanings tend to be more pictographic and are thus easier to interpret, that is, they are more transparent. Representing abstract meaning is more difficult and the meaning of these symbols is dependent on the symbol gloss (the written label associated with the symbol). While traditional literacy allows for the representation of meaning in the form of written words, children who grow up with no speech seldom acquire proficiency in reading and writing. Wilkins and Ratajczak (2009) demonstrate that the use of high-tech AAC devices support the development of literacy skills in users with communication disabilities. One reason for poor literacy attainment is the role that phonological awareness plays in literacy acquisition. For example, English orthography represents the spoken word and children are taught to “sound out” words to write them. Systems such as Blissymbolics provide an alternative to sound-based written languages and can be used by non-speaking individuals who have not developed traditional literacy (Isaacson & Lloyd, 2015). Such systems can

provide users with a means to represent meaning by combining concepts to represent meaning.

## **2.2 Access to AAC Systems by Users with Physical Impairments**

The means of controlling an AAC communication device varies from user to user, depending on the user's physical and cognitive abilities. There are two categories of access methods in terms of selection in an AAC device: direct selection and indirect selection. Direct selection is selecting a target directly with a body part or adapted tool, such as pointing, touching, a computer mouse, or eye-gaze. Indirect selection uses a reduced input such as a binary switch to select a target from a set of choices by scanning each choice in the set. Indirect selection may be used when the user does not have enough physical ability to use a direct selection method.

### **2.2.1 Pointing Devices**

For computers and high-tech communication devices, there is a range of pointing devices. The most recognized device is the computer mouse, though some people with physical impairments are capable of using fingers to use their touchscreen AAC device. However, for users with limited physical movements, there is a range of pointing devices, including the tracker-ball and joystick. The users can operate these pointing techniques to access a system with buttons or keyboard to communicate with others.

### **2.2.2 Switch Scanning**

Switch scanning is an access method that allows people with severe physical impairments to use a computer or an AAC device (Koester & Simpson, 2017). The most popular scanning implementation is row-column scanning, which can be used with one switch for input. In the typical use of a single switch the user can select an item from a two-dimensional matrix with just two switch hits. Each row of the matrix is highlighted in turn until the first switch hit is made to select the appropriate row.

Then each column of the row is highlighted in turn until the user selects the target by doing the second switch hit. This process could be done with a two-switch alternative. One switch is used to move the highlight in their own speed and the other switch is used to select the target in the matrix. As well as row-column scanning there are other scanning techniques such as group-row-column scanning, linear scanning etc.

There are different types of switches that can be used in scanning access method. For users who are not able to use a simple switch, there is a wide variety of alternative switches available, such as eye blink switches, muscle movement switches and sip/puff switches. One of the best-known users of switch access scanning was Professor Stephen Hawking who accessed his computer-based communication system (Hawking, n.d.) by an infrared switch that was mounted on his spectacles. The switch detected Professor Hawking's cheek movement.

### **2.2.3 Eye-Gaze**

Eye-tracking is a technique used to determine where the person is looking. Typically, this is done with an infrared camera that tracks the user's eyes. Eye-gaze systems allow users with physical impairments to access a computer or AAC device, by simply looking at the screen. One of the main issues of an eye-gaze system is that it can be hard to focus on the target on the screen, however research has shown that over time the accuracy improves (Borgestig, et al., 2016).

## **2.3 Natural Language Processing in AAC**

Regardless of the direct or indirect access method, writing for people with physical impairments is challenging. Different Natural Language Processing (NLP) techniques have been used to enhance the usability and the productivity of AAC systems. Newell, et al. (1997) provided an overview of the benefits of research collaborations of AAC and NLP. Over the past few decades there has been significant progress in the use of NLP in AAC systems (Higginbotham, et al., 2012). NLP involves algorithms that modify, augment, or generate human language and methods that range from techniques like word prediction or completion to sentence simplification. The main applications using NLP algorithms focus on language translation, extracting data from large

language corpora, for speech recognition and processing of spoken language (Higginbotham, et al., 2012).

NLP has been used in AAC in many different ways such as text input methods, word prediction, speech recognition and context processing. The design of the “QWERTY” keyboard happened with the invention of the first commercially successful typewriter<sup>1</sup> in 1873 that was designed by the American inventor Christopher Latham Sholes with the help of Samuel Willard Soulé and Carlos Glidden. The QWERTY design attempts to maximise the typing speed by using statistics from English language study, such as by placing the frequently co-occurring characters on opposite sides of the keyboard, which helps the user to type using both hands (Higginbotham, et al., 2012). Changes in keyboard layout have been suggested to improve the typing efficiency with use of statistical properties of the written language (Ichbiah, 1995). Optimising dynamically arranged layouts of the keyboard or keyboards may increase the text input rate and minimise the total motor activity; the unfamiliarity of the layouts has also been an issue for their usage (Higginbotham, et al., 2012). Away from the traditional and ambiguous keyboard layouts, systems like Dasher have shown us how language modelling can be used to help AAC users to write at a faster rate with the use of devices such as a (Higginbotham, et al., 2012). Away from the traditional and ambiguous keyboard layouts, systems like Dasher have shown us how language modelling can be used to help AAC users to write at a faster rate with the use of devices such as joystick or eye-tracker (Ward, et al., 2000).joystick or eye-tracker (Ward, et al., 2000).

As well as in different text input methods, NLP has also widely been used in word prediction. Word prediction is a feature that exists in some form on all high-tech AAC devices. Nearly all these systems use some form of n-gram prediction, one of the most popular methods of word prediction (Higginbotham, et al., 2012). In n-gram the previous  $n-1$  words are used to predict the current word ( $n^{th}$ ) that the user is typing. This data is collected by searching each unique  $n$  word sequence in a large corpus,

---

<sup>1</sup> [https://en.wikipedia.org/wiki/Sholes\\_and\\_Glidden\\_typewriter](https://en.wikipedia.org/wiki/Sholes_and_Glidden_typewriter)

which is called the training text (Leshner, et al., 1999). Trnka, et al. (2007) conducted a study comparing communication rate on basic prediction, advanced prediction and with no prediction. The advanced prediction method used *unigram* to predict the first word of the sentence and *trigram* to predict rest of the words. Even though the study was conducted using subjects with no motor impairments, they simulated a motor impairment by providing an interface that emulates the communication rate of a typical AAC user. The results of the study suggested that the use of word prediction in an AAC device with simulated AAC users significantly enhances communication rate.

## **2.4 Blissymbolics**

This section gives an introduction to Blissymbolics, its invention and how it came into the Augmentative and Alternative Communication world.

Blissymbolics (also known as Bliss) is an ideographic written language that was developed by Charles K. Bliss (1897–1985) for international communication. Bliss can be described in terms of Bliss-characters and Bliss-words. The Bliss-alphabet is comprised of 28 basic shapes, which function as “letters” for the purpose of text sorting and matching operations. Some of the shapes (termed “glyphs” in the context of typography) are stand-alone Bliss-characters while other shapes are segments or fragments, called “strokes” following terminology used in association with Chinese logographs. These elements are used in the design of the Bliss –characters, though in a text-based system the Bliss-characters are not dynamically composed of them. Some strokes can occur in several orientations, sizes, and positions to give different meaning to the characters. All Bliss-characters are defined using the Bliss-alphabet. A Bliss-word can be a single Bliss-character with a specific meaning or a sequence of Bliss-characters with a specific meaning. Sentences are written with these Bliss-words. Currently the Bliss dictionary contains more than 6000 Bliss-words.

### **2.4.1 Background**

Charles K. Bliss was born in Austria in 1897. In 1940, during World War II he and his wife Claire moved to Shanghai, China where he became fascinated by Chinese writing. He started learning Chinese characters. As he learned new characters, he started to

recognize the characters on shop signs or on newspaper headlines and he read them off in his own language, not in Chinese (Okrent, 2009). The Chinese symbol for “person” looks like a person – “人”, and this idea that a character would look like what it is representing fascinated him. He thought that he had discovered a universal language that could be used everywhere, a language mainly based on concepts.

Bliss’s experience of war led him to believe that words were the root cause of many problems in the world. So when he saw the Chinese characters and recognised how similar they looked to the meaning of the symbol, he came upon the idea of inventing a symbolic language which could be used as a universal language – a system which would be modern, simple, and scientifically constructed, a pictorial symbol language which could be typed on a typewriter. Ideally, anyone who looked at the symbols, no matter what their native language, would understand what the symbols meant. Bliss thought that such a language would be the answer for most of the world’s problems, such as language barriers.

Chinese characters, however, are more complicated and most of the characters do not look like the meaning of the character as he thought in the beginning. After a year of studying Chinese characters, he gave up and decided to work on his new universal symbolic language.

At the end of the war Charles and his wife Claire moved to Sydney. He felt that the Australian scholars and University of Sydney would be interested and want to work on this idea of a universal language (Okrent, 2009). But no one was interested. Charles decided to work on his own and after working on the language for seven years, Charles Bliss published a book, *Semantography (Blissymbolics)* in 1949, in which he explained his universal written language: “A logical Writing for an illogical World”. The book contains the rules of the language, meanings of each symbol and how they could be used in the real world.

In his book he wrote,

*“Today blind mankind write in Braille.  
A day will come when seeing and thinking mankind will write and think in Bliss.”  
“A writing which can be read in all languages”  
“A writing which can give literacy to all”  
“A writing which can expose illogic and lies”  
“A writing which can damask and demagogues” (Bliss, 1965)*

After completing the book, he and his wife sent 6000 letters to universities, professors, government officials, and heads of states all over the world announcing the publication of his invention. But there were no responses from anyone.

## 2.4.2 Blissymbolics in AAC

In the 1970s, Blissymbolics become popular in a disabled children’s school, where Shirley McNaughton worked as a teacher. She was one of a group of teachers and therapists who worked with children with Cerebral Palsy. Some of the children were non-speaking and the only communication method they had was pointing to pictures such as food, a drink, a bed, etc.

During the time Shirley McNaughton was working with these children she came across the book that Charles K. Bliss had written. She and the group started working on the Bliss symbols and came up with a chart that contained essential symbols that the children needed to communicate. The chart had nouns (I, he, she, mother, father, etc.), adjectives (happy, sad, frustrated, etc.), verbs (want, eat, understand, etc.) etc. Soon the children started to communicate using symbol combinations and they started to improvise. When one of the children was asked what he wanted to be for Halloween, he first pointed to the symbol for “Creature”, then pointed to the symbol for “Drink”, then the symbol for “Blood” and finally the symbol for “Night”. A Creature Who Drinks Blood at Night is of course a Vampire.

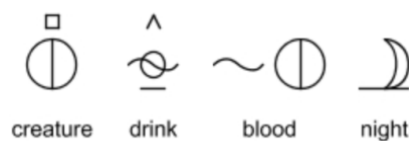


Figure 1: A Creature Who Drinks Blood at Night



In time Blissymbols were developed into a practical language that could be used by users with physical and communication impairments. Bliss became one of the first Augmentative and Alternative Communication (AAC) graphic representational systems, that was used by both adults and children. In 1975 Blissymbolics Communication International (BCI) was established, led by Shirley McNaughton in Toronto, to promote and support the development of the language for disabled people. BCI was granted the licence and copyright to make use of Blissymbolics in their publications. The vision of the BCI was to provide training to professionals to introduce Blissymbolics to people with speech and physical impairments in different countries. The United Kingdom, Australia, New Zealand, Sweden, and Germany became the first countries to undertake training in this application of Blissymbolics.

## **2.5 Fundamental Rules of Blissymbolics**

This section explains the rules and how Blissymbolics is used as a written language.

In 1992 Blissymbolics Communication International (BCI) published the Blissymbol Reference Guide with 2300 vocabulary items and detailed rules for the graphic design of additional characters (Wood, et al., 1992). In the reference guide the bitmap font designed by Peter Reich was used for the printing (Everson, 2020). In 1993 a two-byte graphic character set with 2304 Bliss-words that appeared in the Bliss Reference Guide was registered with International Organization for Standardization (ISO, 1993). This ISO registered character set contained Bliss-words, encoded as discrete lexical items, which became a maintenance issue as new Bliss-words became available/created. Currently there are more than 6000 words in the Bliss vocabulary.

In 1998 Michael Everson produced a new alphabet-based proposal to add Blissymbols the Universal Character Set (ISO/IEC 10646) and its industrial counterpart, Unicode (Everson, 1998). This is different from the existing character set that was registered in 1993. Instead of encoding Bliss-words lexically, Everson's proposal used a Bliss-character model where words are spelled by sequencing their parts. This reduced the size of the character set to about 1400 characters, significantly increasing the flexibility of the writing system. A Bliss font will contain the Bliss-characters which can combine

with one another into Bliss-words along with indicators which will serve to give tenses, verbs, identify plurals, and to differentiate nouns from verbs and adjectives.

With the 1998 Unicode character set proposal came a new document – “The Fundamental Rules” – which set out a comprehensive set of guidelines ensuring compatibility of Blissymbolics with the Universal Characters Set (UCS) (Blissymbolics Communication International, 2009). The rules introduced the concept of Bliss-characters and Bliss-words and specified how Bliss should be written in an alphabet-based system, including aspects of typography such as kerning.

### 2.5.1 Bliss Alphabet



The Bliss-alphabet has 28 shapes and makes use of several alphanumeric characters. More complex Bliss-characters are organized as subsets of the Bliss-alphabet. The shapes can occur in several orientations, sizes, and positions to give different meaning to the symbols. Table 1 contains the Bliss-alphabet characters in a prescribed order that users can use to find other Bliss-characters.

⌋ Wavy Line	⌒ Quarter Circle	⊕ Cross
♥ Heart	) Parenthesis	△ Isosceles Triangle
⏏ Cross Hatch	□ Squares	∧ Symmetric acute angle
⌌ Building	▭ Rectangles	∨ Asymmetric acute angle
⌋ Ear	◻ Open Square	— Horizontal line
↑ Arrow	▭ Open Rectangle	Vertical line
⊗ Wheel	△ Right Triangle	/ Slanted line
○ Large Circle	· Dot	／ Diagonal line
◦ Small Circle	∧ Right Angle	
⌒ Half Circle	⊥ Line on a Base	

*Table 1: Bliss- Alphabet Letters*

## 2.5.2 Bliss-Characters

The Blissymbol character set currently comprises more than 1400 characters and is currently being prepared by Michael Everson as a revised proposal for encoding in ISO/IEC 10646 and Unicode. The Bliss Fundamental Rules document defines Bliss-characters in terms of different types of Bliss-characters; arbitrary; ideographic; pictographic and composite. In the Unicode proposal all the Bliss-characters are organized according to the Bliss alphabet, as they are in the “Finding Symbols by Shape” section of the Bliss Reference Guide. (Wood, et al., 1992) This categorization is used in one of the Bliss-character input methods, which will be detailed in later sections of this thesis. Each Bliss-character could be a single Bliss-alphabet shape or combination of shapes and these shapes can occur in different sizes, positions and orientations.

	Heart - Single Bliss-alphabet shape
	River - Combination of Bliss-alphabet shapes Arrow (horizontal) and Wavy Line (horizontal on the ground level)

*Table 2: Examples of Bliss-characters*

## 2.5.3 Bliss-Words

A Bliss-word can be a single Bliss-character (single-character Bliss-word) with a specific meaning, or a combination of Bliss-characters (multiple-character Bliss-word) with a specific meaning. Bliss-words are separated from one another by one full space or by a Bliss-character half space and a punctuation mark. The characters within a Bliss-word are separated by a Bliss-character quarter space.

🏠	Building (Single Character Bliss-word)
~	Water (Single Character Bliss-word)
🌧	Rain (Single Character Bliss-word)
🏠✉	Post Office – Combination of Bliss-characters “Building” and “Postal Envelope” (Multiple Character Bliss-word)
👤🏠📖	Librarian – Combination of Bliss-characters “Person”, “Building” and “Book” (Multiple Character Bliss-word)

*Table 3: Examples of Bliss-words*

In principle there is no limit on how many Bliss-characters one could use in a Bliss – word (any more than there is in spelling words with the Latin alphabet). There are about 6000 Bliss-words in the Bliss Vocabulary, but Bliss users can also create new Bliss-words outside the Bliss Vocabulary sequencing any of the Bliss-characters.

## 2.5.4 Indicators

Bliss Indicators are markers that are used to indicate grammatical and semantic forms of a Bliss-word. Indicators are divided into Nominal, Verbal, and Adjectival/Adverbial categories. Nominal Indicators specify plurality and definiteness, and the concreteness of a noun that might also have abstract semantics. Verbal and Adjectival Indicators can specify a range of tenses and moods. The Table 4, Table 5 and Table 6 shows the three categories of the Bliss Indicators.

🏠	Thing Indicator
🏠 <sup>x</sup>	Plural Indicator
🏠 <sup>/</sup>	Definite Indicator
🏠 <sup>ox</sup>	Plural Thing Indicator
🏠 <sup>o/</sup>	Definite Thing Indicator
🏠 <sup>x/</sup>	Definite Plural Indicator
🏠 <sup>ox/</sup>	Definite Plural Thing Indicator

*Table 4: Nominal indicators*

	Action Indicator		Past Conditional Indicator
	Active Indicator		Future Conditional Indicator
	Continuous Indicator		Passive Indicator
	Imperative Indicator		Past Passive Indicator
	Present Action Indicator		Future Passive Indicator
	Past Action Indicator		Present Passive Conditional Indicator
	Future Action Indicator		Past Passive Conditional Indicator
	Present Conditional Indicator		Future Passive Conditional Indicator

Table 5: Verbal indicators

	Description Indicator		Past Perfective Participle Indicator
	Description Before the Fact Indicator		Present Passive Participle Indicator
	Description After the Fact Indicator		Past Passive Participle Indicator
	Present Participle Indicator		Future Passive Participle Indicator
	Past Participle Indicator		Description of Action Indicator
	Future Participle Indicator		

Table 6: Adjectival and Adverbial Indicators

The Table 7 shows some examples of how -Indicators are used in Blissymbolics.

	“Beauty” – with the Description Indicator it becomes “Beautiful”	
	“Food” – with the Action Indicator it becomes “To Eat”	
	“Building” – with the Plural Indicator it becomes “Buildings”	
	“Feeling” – with the Thing Indicator it becomes “Heart”	
	“Run” – replacing the action indicator with the Past Action Indicator it becomes “Ran”	

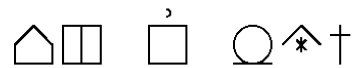
Table 7: Examples of Bliss Indicators use

“The fundamental rules of Blissymbolics” document which was published in 2009 (Blissymbolics Communication International, 2009) as an updated version of the 1999 document. It contains all the information about guidelines for the development of

Blissymbolics, and how this semantic written language can be used as a Universal Character Set with ISO and Unicode.

### 2.5.5 Bliss Sentences

Just like in English and other languages, Bliss sentences are made of Bliss-words. Since Blissymbolics is an ideographic written language, when writing sentences, the Bliss user only has to write the Bliss-words that are necessary to get the concept across. The following example shows the Bliss sentence for “The library was closed on Christmas day”



The word-for-word translation from Bliss to English becomes “Library Closed Christmas Day”. As the Bliss user only has to use the Bliss-words to indicate the concept of that the “The library was closed on Christmas day” the Bliss user can write this in just three Bliss-words.

When asking a question in Bliss you have to specify that you are asking a question in the beginning of the sentence, so the reader knows that it is a question (Bliss, 1965). The following is example of Bliss sentence with a question.



This Bliss sentence translates to “Do you like tea?”. The first Bliss-character of the sentence is “Question” ( $\square?$ ), so the word to word translation from Bliss to English becomes “Question You (to) Like Tea”. The Bliss-character “Question” could mean any form of a question.

## 2.6 Blissymbolics and Technology

This section explores different technologies that are developed for Blissymbolics.

Blissymbolics was one of the first AAC systems to be supported on early 1980s' computer platforms. "Talking BlissApple" was developed for Apple II computers by Gregg Vanderheiden et al at Trace R&D Centre, Madison, Wisconsin, (McNaughton, 1998). Bliss-words were accessed by the user by typing in a number. The users typically had a physical Bliss board with numbers for each Bliss-word. By typing in sequences of numbers, Bliss-words were displayed on the screen and could be printed on to paper.

Current AAC systems, such as Dynavox and Minspeak, which support Bliss tend to treat Bliss as a symbol set. Nowadays Bliss instructors are also able to access Bliss-words for teaching and display purposes via BlissOnline to create Bliss charts and to search for Bliss-words.

### **2.6.1 Bliss-Talk**

Bliss-Talk is an electronic Bliss communication board which was developed by Hunnicutt (1984). This allowed Bliss users to select Blissymbols on an electronic board using a magnet. The selected symbol sentence was then translated into grammatically correct Swedish, English, and French which was spoken via a built-in speech synthesizer. The system could also print the output as needed. The Bliss-Talk communication board had 504 squares, most of which were lexical items (Bliss-words) that were arranged according to the standard Swedish Bliss chart (organized according to parts of speech). It also had squares for the Latin alphabet, numbers, and morphological indicators. Users could reprogram and personalise the board according to which lexical items they needed in their daily conversations.

The Bliss-Talk communication board also allowed the user to indicate the tense of a verb and the plurality or non-plurality of a noun, so the speech synthesizer would output the correct word. In terms of how the system was operated, the Bliss-Talk system had a few advanced features for that time. As the user selected Bliss symbols, the speech synthesizer would output the speech of each symbol meaning, and when the user typed a terminal punctuation mark, the system would output the complete grammatically formed sentence. If the user deleted the last entry the systems would pronounce the string of words up to that point, so the user could make the correction.

Even with its advanced features, Bliss-Talk has some drawbacks. The system is limited to its stored lexical items. As the Bliss-Talk stores Bliss-words, the user can only select from the limited number of Bliss-words stored, which then translate into grammatically correct sentences. This is not a genuine writing system, because the user cannot write anything outside the vocabulary – there is no way they can write anything novel. The language model in Bliss-Talk is a rule-based language model. Even though the Bliss-Talk board can be reprogrammed with new Bliss-words, the new Bliss-words would have to manipulate the lexicon to capture different morphological forms of the word, be it a noun or a verb, adjective, or adverb.

### **2.6.2 BlissOnline.se**

BlissOnline<sup>2</sup> is a website that is organized by the Special Education School Authority in Sweden, which is the largest knowledge base for specialist education in Sweden. BlissOnline allows Bliss users, communication partners, therapists, and teachers to create Bliss-related materials, such as Bliss communication boards (both standard and personalized). The website is primarily used by Bliss communication partners and teachers. The BlissOnline system has a Bliss “Writer” mode which allows users to access Bliss-words by writing the English gloss of the Bliss-words. Since the BlissOnline system only allows users to access stored Bliss-words in the “Writer” mode, users cannot write anything outside the defined Bliss vocabulary.

### **2.6.3 Bliss-Dasher**

Bliss-Dasher provides access to the Blissymbolic language within the Dasher software (Ward, et al., 2000). Dasher is a text-entry interface driven by continuous two-dimensional gestures, accessed, for example, via a mouse, touch screen, or eye-tracker. The Bliss-Dasher system allows the user to steer through the Bliss-alphabet assuming that the user knows the semantic route of the character, and outputs the Bliss prediction.

---

<sup>2</sup> <https://www.blissonline.se/>



Without any training text, Bliss-Dasher assigns all Bliss-characters equal probability. MacKay, et al. (May 2007) states that an able-bodied user can write Bliss at one Bliss-character per 4.5 seconds using the mouse to steer through the matrix. As writing in Bliss-Dasher gets faster with appropriate training text (the appendix of the (MacKay, et al., May 2007) report) the same user was able to write a 12 to 14 Bliss-character phrase from the training set in 4 seconds. Even though the report does not test the system with Bliss users, the comparison of success both with and without a training-text demonstrates that the able-bodied user can achieve a significant increase in the rate of communication.

As this was a prototype system, it contained 500 Bliss-characters. These 500 Bliss-characters were organized partly by semantics and partly by shape. The Bliss-Dasher system has 507 characters in its “alphabet”, which contains Bliss-characters, numerals and Latin letters. These characters can be subdivided into groups, subgroups, and so forth, to a variety of depths. The characters are grouped according to their shape, and each group is ordered by the semantic meaning of the characters (alphabetically by English gloss).

The Bliss-character grouping method of the Bliss-Dasher is complicated. The Bliss-characters are not part of the Bliss-alphabet (Bliss, 1965). Considering there are more than 1200 Bliss-characters it would be difficult to organise the characters in a similar way. The input method of the Bliss-Dasher system can be quite difficult for a user who is not capable of using a joystick or eye-tracker.

## **2.6.4 MisterBLISS**

MisterBLISS<sup>3</sup> is a simple, intuitive, and personalised Bliss writing environment which was developed in 2010. The software is an Italian web-based system, which allows the user to create and edit documents in Bliss through the selection and modification of the individual symbols. Users can access Bliss-words from the Bliss vocabulary by selecting the Bliss-alphabet in the left-hand panel. The user can select multiple Bliss-

---

<sup>3</sup> <http://www.misterbliss.it/>

alphabet letters to filter the Bliss-words that are output by the system. The user also can search for Bliss-words by typing an Italian gloss.

Even with all the features, the MisterBliss program does not allow the user to access Bliss-characters individually, so the Bliss user is unable to compose new Bliss-words and is restricted to what is in the Bliss vocabulary. The system is in Italian, so the system can only be used by users who are fluent in Italian.

### **2.6.5 Simplipedia**

Simplipedia is a system that converts Wikipedia pages into Bliss. This was developed by a MSc student at the University of Gothenburg in Sweden (Tillander, 2014). Simplipedia uses the “CCF-Symbol Server” of the Concept Coding Framework (Lundälv & Derbring , 2012) that links symbols and pictures with mainstream lexical and other Language Technology resources. Simplipedia also uses a Natural Language Processing tool that was developed by the Natural Language Processing Group at Stanford University.<sup>4</sup>

Simplipedia is Bliss conversion system, that gives access for Bliss users to Wikipedia. In the evaluation of the system (Tillander, 2014) states that on average 75% of words have symbols in a converted article. One of the main issues of the system was that when there are words with multiple symbol suggestions the system could choose the incorrect symbol for the context. This could be solved by implementing a symbol selector with help of NLP tools.

### **2.6.6 Bliss OpenOffice Plug-in**

Blissymbols are supported in LibreOffice/OpenOffice Writer and is offered via the Concept Coding Framework SymbolWriter (CCF-SymbolWriter) extension developed in the European AEGIS project (European Commission’s 7th FP, 2008).

---

<sup>4</sup> <https://nlp.stanford.edu/>

Once the user installs the CCF-SymbolWriter extension, a menu is added to the interface of the word processor which allows users to make changes in the settings. These settings include graphic symbol system preferences and general settings. When the extension is activated, a pop-up window appears displaying all the Bliss symbols for the text written in the document. The Bliss symbols are inserted in the document on top of the text words, as a Private-Use Area Unicode font. When more than one symbol representation is available for a word, the user can select the correct symbol which can then be inserted into the document. Lundälv, et al. (2014) lists the three main benefits of CCF-SymbolWriter as: the support and confirmation of comprehension for symbol users in writing and reading text; giving access to AAC users to the full symbol representation with symbols displayed on top of each word; and finally supporting the helpers (parents, teachers, therapists etc.) to prepare documents with symbol support.

The CCF-SymbolWriter is similar to most of the AAC systems that support Blissymbolics in that it does not enable Bliss users to generate Bliss themselves. Even though the application gives access to the full symbol vocabulary, the user can only access the symbols by typing words into the document. If the user writes a word that is not represented in the Bliss vocabulary, the user cannot access that word in Bliss.

## **2.7 Bliss as a Character Set**

This section investigates character input methods for larger character sets such as Chinese. This section also outlines two Bliss-character input methods.

As described in section 4, Blissymbolics is an ideographic language that has four main aspects: Bliss-alphabet; Bliss-characters, Bliss-words, and Indicators (which are non-spacing Bliss-characters similar to the diacritical marks used in French and other languages). Just as in English we use letters to write words and arrange words to make sentences, in Bliss there are Bliss-characters and Bliss-words, with the Bliss-characters being arranged to the shape classes in the Bliss-alphabet to which they belong.

Currently in Blissymbolics there are about 1400 Bliss-characters. This poses a number of challenges:

- Retrieval of target Bliss-characters within a large character set;
- Retrieval of Bliss-characters by Bliss users with physical impairments;
- Writing Bliss-words by Bliss users with physical impairments.

## **2.7.1 Accessing Bliss-characters**

Bliss is an ideographic language with a large character set. Since Chinese writing had an influence of Charles Bliss in developing Blissymbolics and its large character set, it is worth investigating Chinese character input methods.

### **2.7.1.1 *Chinese Character Input Methods***

Chinese is a language written with thousands of logographic characters. With the age of computing one of the main issues in Chinese computing centres is how users input Chinese characters on digital devices, such as computers, mobile phones, and smartphones. Over the years a variety of Chinese input methods have been developed. These different input methods can be categorised into two main categories sets: i) Phonetic-based, where the user enters pronunciation that is converted into Chinese characters and ii) Shape-based, which is based on how the characters are written, i.e., their shape/stroke (Chen & Lee, 2000). When using a Phonetic-based input method the user needs to know how the Chinese characters are pronounced and when using a Shape-based input method the user should know how the characters are written.

The most widely used Phonetic-based input method in China is the Pinyin input method. It is used by over 97% of users in China (Chen & Lee, 2000). In Pinyin the English alphabet letters are used as phonetic symbols to type Chinese characters. The main advantage of Pinyin method is that it is easy to learn (at least for speakers of Standard Mandarin Chinese), and it only requires the user to know how to speak Chinese and be able to recognise Latin characters. As well as many advantages the Pinyin-based method has a few issues, such as Pinyin to character conversion errors, user input errors and problems with the user interface, e.g. when the user writes in

Chinese and in English, users have to use two separate modes to type (Chen & Lee, 2000). However, these issues have been addressed in research such as, CHIME: an efficient error-tolerant Pinyin input method that was developed by Zheng, et al. (2011). CHIME addresses the issue in user input errors in Pinyin by automatically detecting and correcting the typing errors in Pinyin input sequences. In terms of the input speed the Phonetic-based input method is relatively slow.

A Shape-based input method has a higher input speed. In a Shape-based input method the Chinese characters are structured in terms of radicals (which have a kind of alphabet order) and strokes (which are written in a prescribed order from top left to bottom right). The user can input Chinese characters using fewer keystrokes than the Phonetic-based input method (Po, et al., 2009). The two most popular Shape-based input methods are: Wubi and Cangjie. Wubi is the fastest Shape-based input method, and widely used in mainland China for simplified Chinese characters (Po, et al., 2009). The keyboard is divided into five sections; each section is assigned to a stroke corresponding to how Chinese characters are written by hand. The five strokes are falling left, falling right, horizontal, vertical, and hook. The keys that assigned to these sections are, QWERT, YUIOP, ASDFG, HJKLM and XCVBN (Bluche & Messina, 2016).

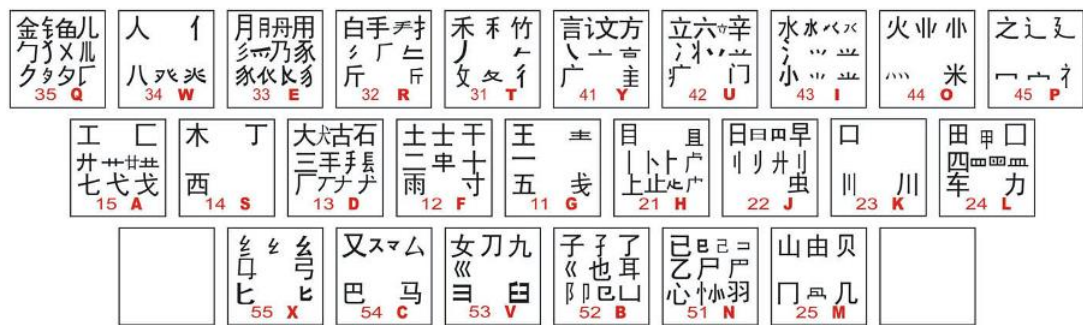


Figure 2: Wubi 86 Keyboard Layout (from Wikipedia)

Users can type the most frequently used Chinese characters with just two keystrokes and a maximum of four. For any character that has more than four strokes or radicals, the user types the first three components, followed by the last one. (Bluche & Messina, 2016). This way the user can type any character in no more than four keystrokes.



Figure 3: Cangjie Keyboard Layout (from Wikipedia)

Cangjie is another shape-based Chinese input method, which is widely used in Taiwan and Hong Kong. Similar to the Wubi method, Cangjie uses the QWERTY keyboard to map Chinese character radicals and strokes. There are 24 basic components/radicals (basic shapes) that are associated with 76 supplementary shapes, which are rotated or transposed versions of the basic shapes. The 24 components are grouped into four subsets. In the QWERTY keyboard the letters from ‘A’ to ‘G’ contains the Philosophical set, the characters representing the sun, moon and the five elements. The keys from ‘H’ to ‘N’ represents the strokes. The keys from ‘O’ to ‘R’ are the human body related components and keys from ‘S’ to ‘W’ and letter ‘Y’ represents the complex character forms. The letters ‘X’ and ‘Z’ are used to special functionality such as to mark a character that is difficult to compose or to produce special characters such as punctuation (Bluche & Messina, 2016).

Since Blissymbolics has no phonology, it is impossible to develop a phonetic-based input method for it. A solution is therefore to develop an input method based on the Bliss alphabet.

### 2.7.1.2 *Bliss alphabet-based Input method*

Bliss-characters have the potential to be accessed using the shapes of the characters. The Blissymbols Unicode proposal (Everson 1998) divides all the Bliss-characters according to the first Bliss-alphabet shape from which they are formed. The proposal suggests how all the Bliss-characters can be categorized by Bliss-alphabet shape. In order to find a Bliss-character, the user begins by looking up the first Bliss-alphabet

shape of the character. Table 8 displays how a subset of Bliss-characters are categorized into the Bliss-alphabet shapes Wavy Line, Heart, and Building.

Wavy line }		Heart ♡		Building 🏠	
}	Raised Wavy Line	♡	Feeling	🏠	Building
}	Fire	💞	Love	🏠	Garage
⌘	Extinguishing	💞	Slang	🏠	Mobile Home
🌀	Mango	💞	Spirit	🏠	Indoor
👤	Vase or Urn	👤	Captivity	🏠	Outdoors
~	Horizontal Wavy Line on Skyline	👤	Freedom	🏠	Building with Attic
≈	Three Channels of Water	👤	Naturalism	🏠	Attic

Table 8: Some Bliss-characters and their associated Bliss-alphabet shapes

Bliss-characters can also be accessed using the Bliss shapes as strokes that are used to form these Bliss-characters. The user accesses Bliss-characters by inputting the Bliss-alphabet shapes to access all the Bliss-characters that contains those shapes. This input method is similar to the Chinese shape-based input methods, Wubi and Cangjie.

## **3 Initial Keyboard Design**

As established in earlier research work, there is no accurate way for a Bliss user to write in Bliss. In the past most Bliss related AAC systems have supported the teacher or therapist rather than the Bliss user. The Bliss user would select Blissymbols from a predefined chart that was configured by the therapist and form sentences which would then output as speech. However, in the past couple of decades, the focus of these systems has shifted towards the Bliss user. The Bliss user has been able to read web articles in Bliss (Tillander, 2014); write in Bliss with the help of AEGIS project (Lundälv & Derbring, 2012) and make use of an email reader and writer for Bliss users (Várady, et al., 2002).

Even though these systems focus on Bliss users, they are focused on literate Bliss users. The Bliss user can use the symbol set only if they understand traditional orthography. There has not been much focus on non-literate Bliss users (MacKay, et al., May 2007). The aim of this research is to develop a system which allows Bliss users to express themselves in Bliss by writing in Bliss, rather than choosing from a pre-set vocabulary on a symbol chart.

The goal of this research is to investigate a system which would allow users to write in Bliss with use of a Bliss keyboard. This could be on a computer or a mobile device such as a tablet.

### **3.1 The Development of the Bliss Writer**

In the past, most systems developed for Bliss users for writing, reading and transition have displayed Blissymbols in bitmap form. This makes it harder to produce documents that are written in Bliss. Blissymbolics is a symbol system that can be drawn by hand on paper, unlike other graphical AAC systems. Bliss is well suited for all AAC technology levels, from no-tech to hi-tech.

When drawing Blissymbols by hand, the strokes of each Bliss-character must be precise. Charles Bliss envisioned using a specially cut stencil (Figure 4) being used.



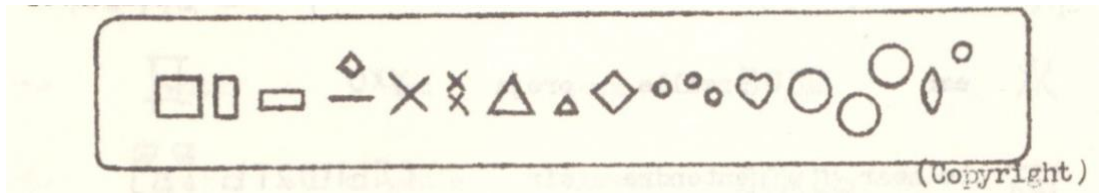


Figure 4: Blissymbol drawing stencil (Bliss, 1965)

The Bliss stencil was developed and came into general use once symbols began to be used by children (Everson, 2020).

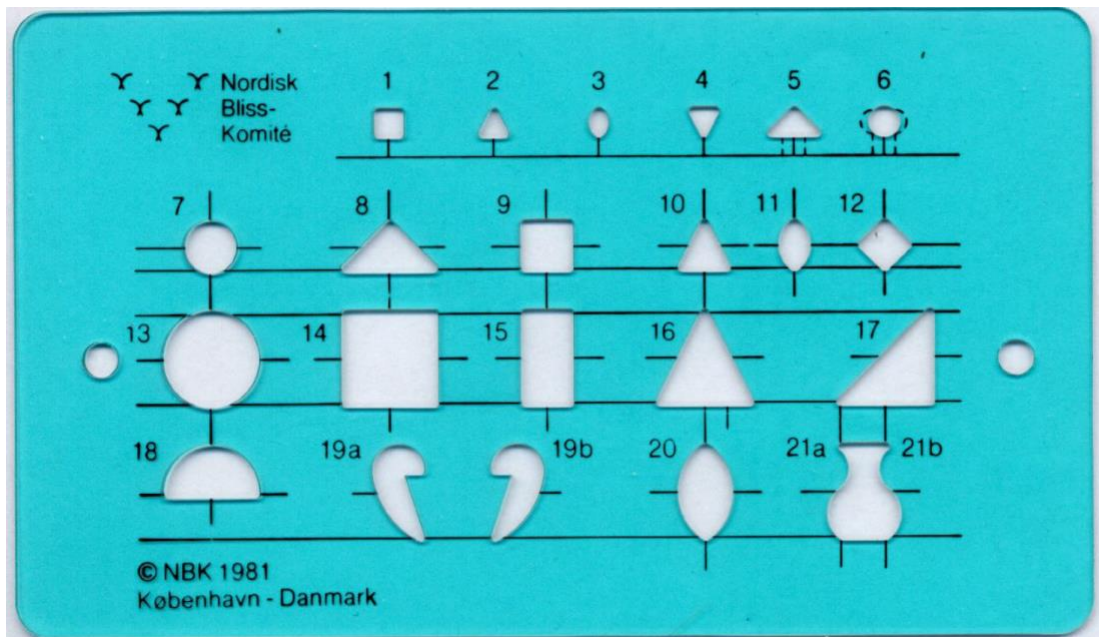


Figure 5: Blissymbol Stencil Template (Everson, 2020)

In Charles Bliss' Semantography book he describes how you could write Blissymbols with a "Semantographic Typewriter" (Bliss, 1965). This is an ordinary size typewriter which shows the usual set of keys with small alphabet letters, numbers and a few punctuation marks like the comma, full stop, question mark, exclamation mark and brackets. With the shift key, instead of capital letters the typewriter gives access to the straight and curved lines which are used to compose the Blissymbols.

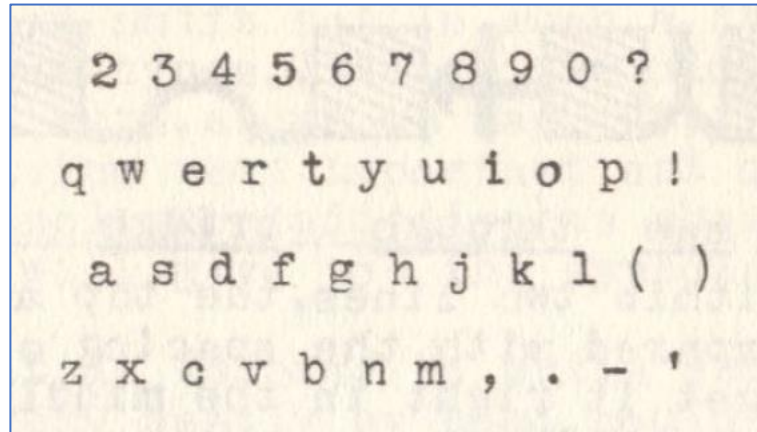


Figure 6: First set of keyboard keys, which we find on any typewriter (Bliss, 1965)

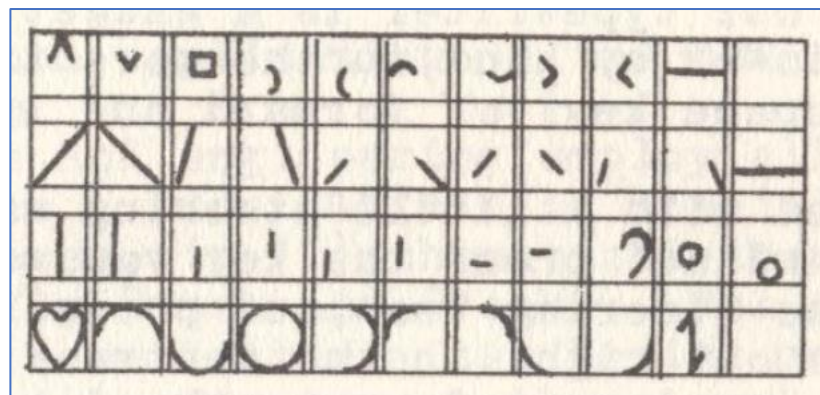


Figure 7: Second set of keyboard keys represent the strokes for composition of the Blissymbols (Bliss, 1965)

To compose Blissymbols with these strokes (Figure 7) using the typewriter, Charles suggests that we may have to add a little gadget to the “Semantographic Typewriter” (Bliss, 1965). In a traditional typewriter, every time a key is pressed the cylinder jumps one space ahead to type the next character. With the Figure 7 keyboard keys if we want to type the Mathematical symbol plus (+) by typing horizontal line (–) and vertical line (|), on a traditional typewriter we would get this – |, simply because the cylinder jumps one space ahead. However, by introducing a small gadget that is operated by a leaver on the typewriter, we could stop the typewriter cylinder jumping ahead. This would allow us to type the mathematical symbol plus (+) with horizontal (–) and vertical (|) lines using the Semantographic typewriter. Any key can be used to compose different Blissymbols and once the Blissymbol is completed, the long space bar is pressed to move the cylinder ahead to the next space.

With the Semantographic typewriter Blissymbols are printed between two lines, the top line and the bottom line. Charles suggests another innovation on the Semantographic typewriter. Rather than moving the typewriter cylinder by hand, a steering stick like a gadget would move the cylinder slightly (Bliss, 1965). It would be held in the left hand and the key would be pressed with the right hand. Moving the steering stick would change the position of the print. By moving the steering stick up or down and pressing any key, it would print on the top or bottom line and similarly if moved left or right and pressing any key it would print slightly left or right to where the key was supposed to print originally. This way different keys can be printed in nine different positions within a square. For example, typing the dot key while using the steering stick prints the dots in following positions (Figure 8):



*Figure 8: Dot printed using Semantographic Typewriter with a Steering stick (Bliss, 1965)*

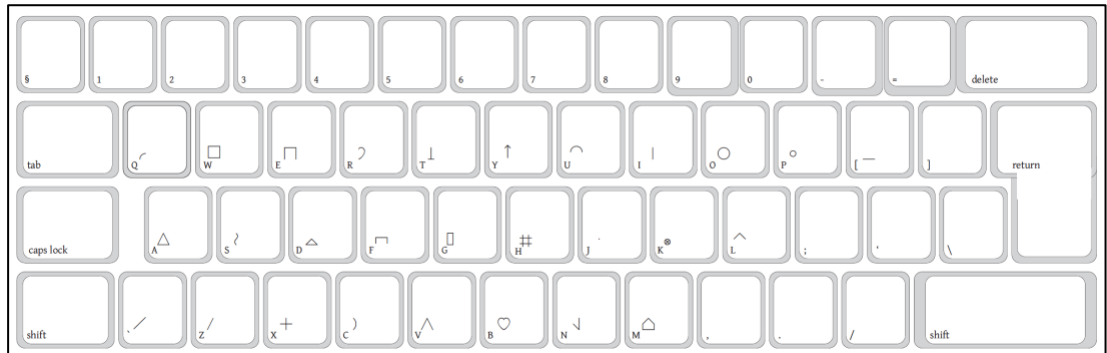
With the addition of the steering stick the Semantographic typewriter allows for the typing of a few more symbols such as “Opening” (⌊) and “enclosure” (⌋) by using the long horizontal (—) and vertical (|) lines.

In high-tech systems, using a font to write Blissymbols in an AAC device or a computer is the most logical solution. A Bliss Unicode font is currently being developed by Michael Everson. To write Blissymbols using the font we need to find a way to write more than 1400 Blissymbols in a modern-day device such as a desk-top computer or Tablet. For this, the Bliss-alphabet shapes were mapped onto the QWERTY keyboard using a similar approach that Charles took when he proposed the idea of Semantographic Typewriter (Bliss, 1965).

### **3.1.1 Bliss-Alphabet Mapping onto the QWERTY Keyboard**

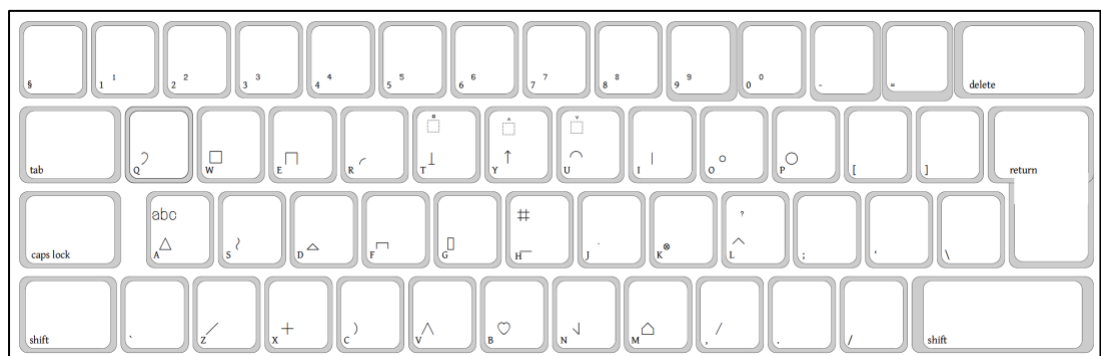
In order to access Bliss-characters using the Bliss alphabet, a Bliss keyboard (Everson, 2020) was initially designed by mapping the Bliss-alphabet to the QWERTY keyboard. The mapping reflected the similarity between the Bliss-alphabet shapes and

the English letters on the QWERTY keyboard. For example, the Bliss-alphabet shape ♡ was mapped to the letter ‘B’ key as it looks like a sideways ‘B’ and the Bliss-alphabet shape ^ was mapped to the letter ‘V’ key as it looks like an inverted ‘V’. The keyboard uses the 26 letters and two punctuation mark keys: the left square bracket ([) and the grave accent (`).



*Figure 9: First Iteration Bliss Keyboard Mapped to QWERTY Layout (2018) (Everson, 2020)*

In 2020 the researcher worked with Michael Everson to update the 2018 Bliss Keyboard. As the two worked together, Everson (2020) made some changes to his version of the Bliss keyboard in order to optimize the positioning of more frequently used Bliss-alphabet keys on the keyboard. An example of these changes includes the swapping of the mappings to the Q and R keys; in 2018 the Ear ♡ was on R (mnemonic the bowl of the R) and the Quarter-circle ◌ was on Q (mnemonic the tail of the Q). This was altered because there are 41 Bliss-characters in the Quarter-circle class but only three in the Ear class, so the access to greater number of characters (41) were moved to the more central position, onto the letter R (Everson, 2020).



*Figure 10: Updated Bliss Keyboard Layout (Everson, 2020)*

A further design modification to the keyboard was made by the researcher to accommodate the Bliss indicators and punctuation-derived class. The three Bliss indicator classes are accessed by pressing the shift key. These can be seen in Figure 10 on the T, Y and U letters. The punctuation-derived class can be access by pressing the shift key, this can be seen in the Figure 10 on the letter ‘L’. One member of the Bliss-alphabet, Cross-hatch, is also found on shift ‘H’ in this iteration of the keyboard layout.

The researcher used this keyboard layout to develop an onscreen keyboard that is displayed within a web application. The user can type Bliss-characters using the onscreen keyboard. **Error! Reference source not found.** and Figure 12 displays the onscreen Bliss keyboard. With the onscreen keyboard layout the researcher added a button to switch between the Bliss-alphabet keyboard and English Alphanumeric keyboard in QWERTY layout (Figure 13). This is located on the bottom-left corner of the onscreen keyboard (“ABC/Bliss” key).

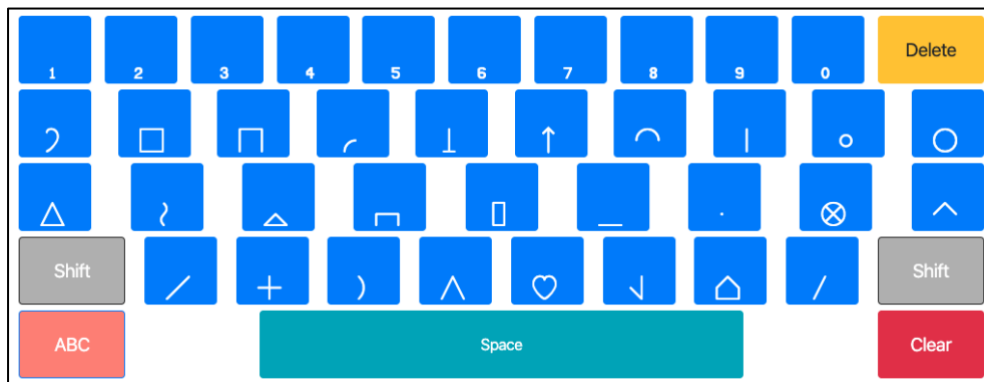


Figure 11: Bliss Onscreen Keyboard

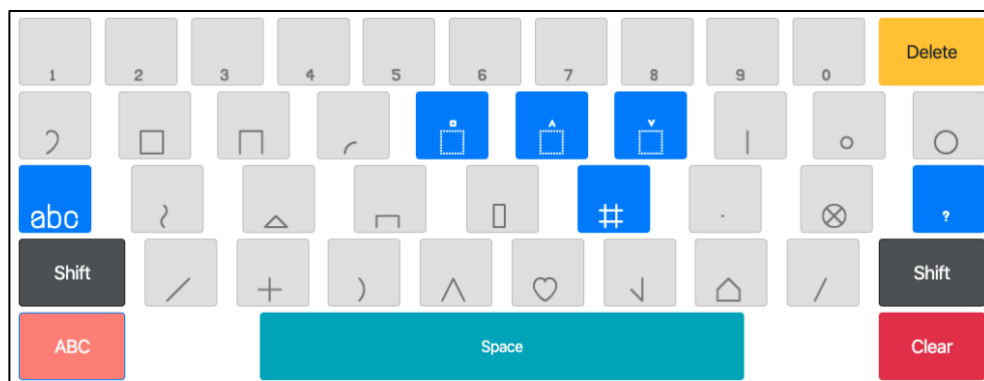
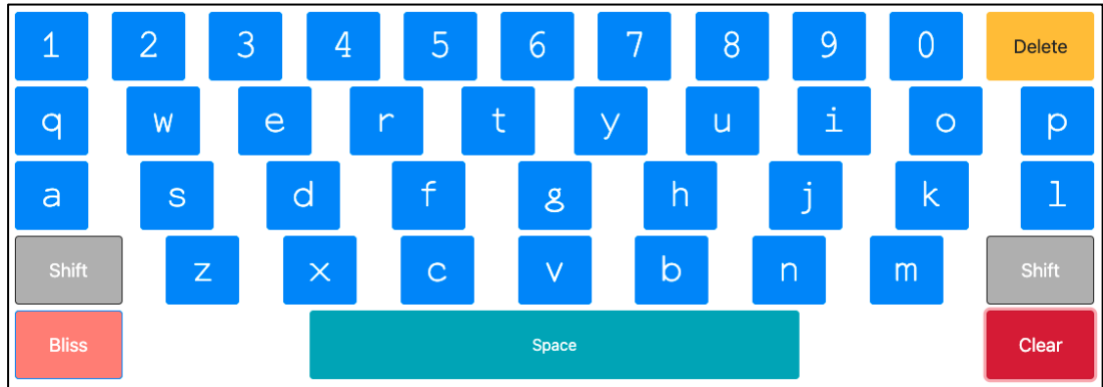


Figure 12: Bliss Onscreen Keyboard (Shift-pressed)



*Figure 13: English Alphanumeric keyboard in QWERTY layout*

This study has investigated two ways to input Bliss-characters using the Bliss alphabet. The two input methods were based on similar systems in Chinese and with reference to the ideas set out in the Blissymbols Unicode proposal (Everson 1998): the Bliss alphabet-based and Bliss stroke-based input methods.

## 3.2 Bliss Writer - System Design

This section discusses the implementation process of the Bliss Writer system.

To develop the Bliss Writer system the following user stories were developed.

### 3.2.1 User Stories

1. As a user I should be able to write in Bliss using the Bliss font rather than Bitmap images, so I can write Bliss using a word processor.
2. As a user I should be able to type Bliss-characters using a QWERTY keyboard that is mapped to the Bliss keyboard so I can access the system with a standard keyboard.
3. As a Bliss user I should be able select Bliss-alphabet letters from the Bliss keyboard, so I can use Bliss-characters to write Bliss
4. As a user I should be able select Bliss indicators with Bliss-characters so that I can express them grammatically
5. As a user I should be able to type the English alphabet using the keyboard, so I can write anything in English

6. As a user I should be able to choose Bliss-characters using the Bliss alphabet-based input method
7. As a user I should be able to choose Bliss-characters using the Bliss stroke-based input method

The Bliss Writer is a web application. The reason behind developing a web application rather than a desktop application is that it is easier to access a web application on any device with a web browser, making it more accessible for users who may use a range of input devices. The web application was developed using the Meteor Web Framework (see Appendix D for details on the implementation).

### **3.2.2 Bliss-character Input methods**

The Bliss Writer system uses an interim version of the Bliss Unicode font developed by Michael Everson (Everson, 2020). The Bliss Writer system allows the Bliss users to write in Bliss using two different Bliss-character input methods:

1. Bliss alphabet-based input method
2. Bliss stroke-based input method

#### **3.2.2.1 *Bliss Alphabet-based Input method***

The first Bliss-character input method to be implemented was the Bliss-alphabet-based input method. The Bliss-character set that Michael Everson was using to develop the Bliss Unicode font was already categorised according to the Bliss-alphabet shapes. So, the researcher was able to generate the database with that data. The Bliss user can access Bliss-characters by selecting the Bliss-alphabet shape and the Bliss user is then presented with all the Bliss-characters under that shape. Figure 14 shows an example of how the Bliss-alphabet-based input method has been implemented as the Bliss-alphabet keyboard.

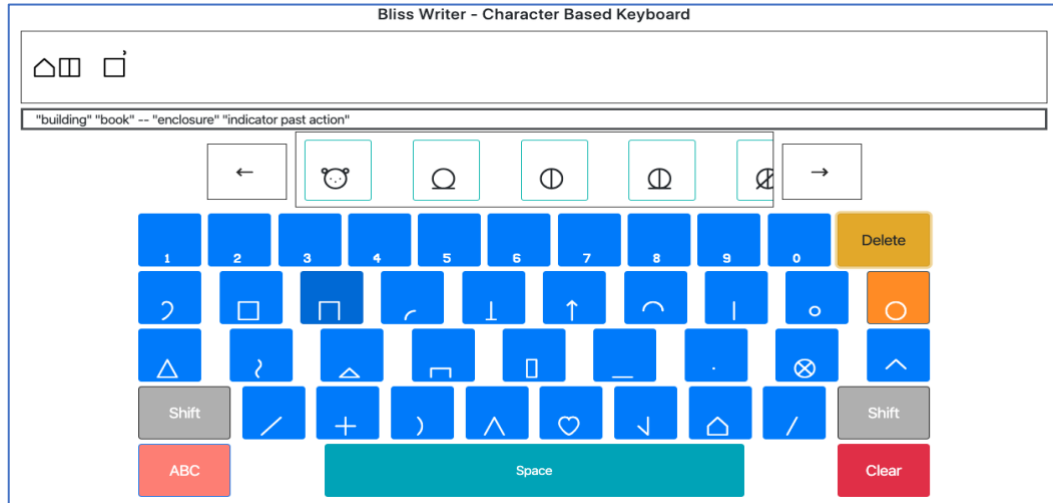


Figure 14: Bliss Writer – Alphabet-based Keyboard

### 3.2.2.2 *Bliss Stroke-based Input method*

As well as Bliss-characters, the full Bliss-character set also contains indicators, punctuation, Bliss digits, currency signs and fractions. For the purposes of this project, the Bliss stroke-based input method analysis has only been done with the Bliss-characters that are categorised by Bliss-alphabet shapes.

The Bliss stroke-based input method was implemented after doing an analysis on how each Bliss-character is formed in terms of the Bliss-alphabet shapes. Since there was no data on this, the researcher had to do this analysis by going through all 1236 Bliss-characters and identifying what Bliss-alphabet shapes are used to create each Bliss-character. As each Bliss-character is formed by Bliss-alphabet shapes, in the Bliss strokes analysis, each character is broken into their Bliss-alphabet components. In each Bliss-character the stroke components are sorted by the Bliss-alphabet. However, when a Bliss user accesses a Bliss-character using the stroke-based input method, they do not need to know the order of the strokes. They are able to access Bliss-characters by inputting Bliss-alphabet shapes in any order and can access any Bliss-characters that have those Bliss-alphabet shapes. Table 9 shows a few of the Bliss-characters from the analysis.














Bliss-character	Bliss-character Name	Bliss Strokes in Bliss-alphabet Shapes
	TENNIS	o o + + +
	EYEGLASSES	o o ) . .
	SWAN	^ } } /
	APPLE	^ ^ ^ ^
	CURTAINS	^ ^ □
	Hail	} □ ↑
	Lorry	⊗ ⊗ ⊗ □ ⊥ _
	Head with Hair	○ ⊥ ^ ^ ^
	Chest of Drawers	□ . . _
	Mother	^ △

Table 9: Few Bliss-characters from Bliss Stroke-based Input method Analysis

This analysis data was added to the Bliss-character database in the Bliss Writer system to facilitate stroke-based access. The design implementation of the Bliss stroke-based keyboard is similar to the Bliss alphabet-based keyboard. However, as the user types multiple Bliss-alphabet shapes using the keyboard, the list of Bliss-characters that are displayed gets smaller. For example, if the Bliss user wants to access Bliss-character “helicopter”  (Figure 15) they can access that more quickly just by typing the “Wheel” and “Slanted line” Bliss-alphabet shapes. If the user is to find the Bliss-character “helicopter” using the Bliss alphabet-based input method, they would have to go through many more Bliss-characters after selecting the “Wheel” Bliss-alphabet shape since the “helicopter” is located at the end of the Bliss-alphabet “Wheel” category Bliss-character set.

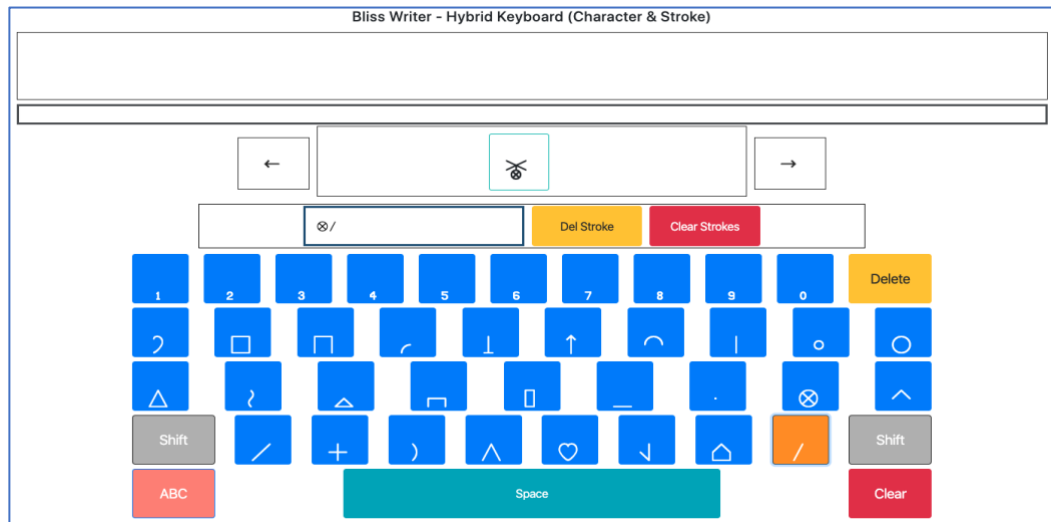


Figure 15: Bliss Writer – Stroke-based Keyboard

## 4 Study 1

This chapter describes the Study 1 participants, methodology, results and the analysis. The participants were given two sets of Bliss sentences to copy using the two Bliss-character input methods: Bliss alphabet-based and stroke-based. The participants used the Bliss Writer system to copy type the sentences and the results were analysed to answer the first two research questions:

*RQ1. Is it possible to design a keyboard input method for the Bliss-character set?*

*RQ2. What is the experience of people with physical impairments when accessing Bliss-characters using different access methods?*

### 4.1 Participants

The participants were recruited through a volunteer AAC User Group. Ethical approval was obtained through the SSEN Ethics Committee (Appendix B). Seven participants consented to participate in the study. Only two of the participants had knowledge of Blissymbolics. Table 10 summarises the participants and their access methods.

Participant	Access method	Bliss Knowledge
P1	Touch	Has fairly good knowledge.
P2	Touch	None
P3	Touch	None
P4	Eye-Gaze	None
P5	Joystick	None
P6	Touch with a Keyguard	None
P7	Touch	Has fairly good knowledge.

*Table 10: Study Participants, Access Method and their Bliss knowledge*

## 4.2 Method

All the study participants were sent the information sheet and the consent form prior to the study. The study was conducted as a two-hour group session in a University seminar room with a multimedia projector. On the day the researcher provided information sheets and consent forms to any participants who had not yet signed.

Since most of the participants did not have any knowledge of Blissymbolics, a Bliss introductory training session was conducted by Professor Annalu Waller. Professor Waller presented the concepts of the Bliss alphabet, Bliss-character and Bliss-word. This also included how Bliss-characters are formed using the Bliss-alphabet shapes. After the 15-minute Bliss training session, the researcher introduced the Bliss Writer system which the participants would be using in the study. The researcher described the two Bliss-character input methods (alphabet-based and stroke-based) to the participants and the introduced the Bliss keyboard layout which the participants would use to access Bliss-characters. Even though most of the participants did not have any Bliss knowledge, it was important to make sure the participants understood the two Bliss-character input methods. Participants did not need to understand Blissymbols as they were able to access the Bliss-characters using the Bliss alphabet-based and stroke-based input methods to copy the Bliss sentences provided.

Participants were given their login details along with a list of sentences to copy in Bliss after the Bliss Writer introduction. Once each participant had logged onto the system, they were presented with two options: the Bliss alphabet-based keyboard which allows the user to access Bliss-characters using the Bliss alphabet-based input method, and the Bliss stroke-based keyboard which allows the user to access Bliss-characters using the Bliss stroke-based input method.

First the participants were asked to copy seven sentences using the Bliss alphabet-based keyboard. Once they finished copying the seven sentences using the Bliss alphabet-based keyboard, they were given another seven sentences to copy using the stroke-based keyboard.

The Bliss sentences use a variety of Bliss-characters. The participants needed to use different Bliss-alphabet keys and Bliss indicators to copy these sentences.

### **Bliss Alphabet-based Keyboard Sentences**

1.  $\dot{\perp}_3 \dot{\rightarrow} \sim \square \times \sim$  – They swam in the lake
2.  $\perp_{1+} \dot{\otimes}_1 \pi \zeta \pi$  – My favourite animal is dog
3.  $\perp_3 \hat{\heartsuit} +! \hat{\triangle} \rightarrow!$  – He likes to run
4.  $\lfloor \rfloor \perp_1 \hat{\downarrow} \sim$  – Can I have some water?
5.  $! \heartsuit \# \perp_{2+} \# \div$  – Please wear your coat
6.  $\lfloor \rfloor \perp_2 \hat{\heartsuit} +! \otimes \emptyset$  – Do you like tea?
7.  $\perp_{1+} \hat{\uparrow} \hat{\rightarrow} \perp_1 \mathbb{D}_4$  – My mother is coming to see me in April

### **Bliss Stroke-based Keyboard Sentences**

1.  $\Delta_3 \heartsuit \} \hat{\otimes} \square \hat{\square}$  – She wants to buy books
2.  $\lfloor \rfloor \mathbb{Q}_1 \perp_2 \hat{\triangle} \triangle \hat{\uparrow} \square$  – Are you going to school tomorrow?
3.  $\perp_1 \dot{\perp} \circ \boxminus \perp_{1+} \hat{\otimes} \nearrow$  – I have lost my phone
4.  $\triangle \square \square \mathbb{Q} \hat{*} \dagger$  – Library was closed on Christmas day
5.  $\dot{\perp}_1 \circ \theta \otimes \rightarrow$  – We can watch a film
6.  $\triangle \square \square \mathbb{Q}_1$  – Post office was closed on Sunday
7.  $\dot{\perp}_3 \dot{\rightarrow} \sim \square \times \sim$  – They swam in the lake

During the session, the researcher and his colleagues assisted the participants if they had any questions. These questions included technical questions such as login to the system and selecting the correct Bliss-character input method keyboard to copy the sentences. Some participants asked for assistance on accessing specific characters. This is due to participants not having much knowledge on Blissymbols. On these

occasions the researcher and his colleagues guided the participant on how to access the specific Bliss-character.

No observational data was gathered during the study. The Bliss Writer system logged all participant interactions. Every keystroke and selection from the participants were logged. The following log data was gathered during for study 1.

1. Data and time of each interaction
2. Participant Username
3. Bliss-character input method
4. Log event
5. Bliss keyboard input
6. Bliss-character the participant selected
7. Bliss-character meaning in English (participant selected)
8. Unicode code of the Bliss-character (participant selected)
9. In the stroke-based input method, the strokes the participant typed to access the Bliss-character
10. Bliss-character list and the number of Bliss-characters presented to the participant on each keystroke of Bliss keyboard
11. Bliss-characters in the Bliss Writer text box

## **4.3 The Bliss Writer Study 1 Results**

This study was done with seven participants with various physical and speech impairments. Most of the participants did not have any knowledge of Blissymbolics. The participants were asked to use two Bliss keyboards with different Bliss-character input methods.

1. Bliss alphabet-based input method
2. Bliss stroke-based input method

Table 11 gives an overview of how each participant performed on each Bliss keyboard.

Participant	Sentences completed in Bliss Alphabet-based Keyboard	Sentences completed in Bliss Stroke-based Keyboard	Length of session
P1	7	0	51 Mins
P2	1	0	1 Hour 5 Mins
P3	7	0	55 Mins
P4	5	0	55 Mins
P5	0	0	30 Mins
P6	2 ½	1	40 Mins
P7	5	2	1 Hour 11 Mins

*Table 11: Study Participant's session details*

Due to the limited time some of the participants only used the alphabet-based keyboard (P1, P2, P3 and P4). However, the log data shows P2, P6 and P7 were able to use both keyboards and only P6 and P7 were able complete a sentence on both Bliss input methods.

### 4.3.1 Participant 1 – P1

P1 was the most enthusiastic participant. The participant expressed delight in being able to write Bliss-words which they recognized despite not having used Bliss for some years.

The participant used an iPad to access the Bliss Writer system and used touch access. Due to the limited time P1 was only able to work with the Bliss alphabet-based input method. They managed to complete all the sentences they were provided. The participant's communication partner did help them in using the system. The communication partner would help in pressing onscreen keys when P1 asked them to do so.

The following table is a comparison of the original Bliss sentences and participant's typed Bliss sentences.

Bliss Alphabet-based Keyboard Sentences	Participant's Bliss Sentences	Notes
$\perp_3 \quad \dot{\rightarrow} \sim \quad \square \quad \times \sim$	$\perp_3 \quad \dot{\rightarrow} \sim \quad \square \quad \times \sim$	The participant got the last Bliss-word “lake” ( $\times \sim$ ) wrong. Even though this looks similar the Bliss symbol for “much” should be smaller ( $\times$ ) and the Bliss symbol “water” ( $\sim$ ) should in ground level.
$\perp_{1+} \quad \check{\theta}_{1.} \quad \pi \quad \zeta \pi$	$\perp_{1+} \quad \check{\theta}_{1.} \quad \pi \quad \zeta \pi$	The participant has not written the description indicator for the Bliss-word “favourite” ( $\check{\theta}_{1.}$ ).
$\perp_3 \quad \hat{\heartsuit}+! \quad \hat{\triangle} \rightarrow!$	$\perp_3 \quad \heartsuit^+! \quad \triangle \rightarrow!$	The participant has typed the wrong indicator Blissymbol. The participant also typed the space between the last Bliss-word “run” ( $\hat{\triangle} \rightarrow$ ).
$\boxed{?} \quad \perp_1 \quad \hat{\downarrow} \quad \sim$	$\boxed{?} \quad \perp_1 \quad \updownarrow \quad \sim$	The participant has written the Bliss-word “have” ( $\hat{\downarrow}$ ) wrong. It seems they have mistaken the action indicator with an arrow pointer.
$!\heartsuit \quad \hat{\pi} \quad \perp_{2+} \quad \pi \div$	$!\heartsuit \quad \hat{\pi} \quad \perp_{2+} \quad \pi \div$	The participant has written the Bliss-word “your” wrong. They have used the middle level plus Bliss symbol (+) instead of the earth line level plus symbol which indicates “belongs to” (+) in Bliss.
$\boxed{?} \quad \perp_2 \quad \hat{\heartsuit}+! \quad \odot \emptyset$	$\boxed{?} \quad \perp_2 \quad \hat{\heartsuit}+! \quad \odot \emptyset$	The participant has not made any mistakes in this sentence.
$\perp_{1+} \quad \hat{\uparrow} \quad \hat{\triangle} \perp_1 \quad D_4$	$\perp_{1+} \quad \hat{\uparrow} \quad \triangle \perp_1 \quad D_4$	The participant has not written the Bliss Indicator action in the Bliss-word “visit” ( $\hat{\triangle}$ ).

Table 12: Participant 1 Bliss alphabet-based Keyboard Sentences



By looking at the output, P1 has typed almost all of the sentences identically with few mistakes. From the seven sentences P1 has managed to copy one sentence identically and other six with only one minor mistake on each sentence.

### 4.3.2 Participant 2 – P2

P2 did not seem to understand the concept of Blissymbolics. P2 does not use any AAC system to communicate as they have good intelligibility. Their disability is acquired and because they have good literacy skills, they did not see the need for an alternative to a literacy-based system.

They were therefore critical about both Bliss keyboard input methods. P2 only completed one sentence using the Bliss alphabet-based keyboard. P2 did try the Bliss stroke-based keyboard but did not complete any sentences due to lack of time. The following table is a comparison of the original Bliss sentences and the participant's typed Bliss sentences.

Bliss Alphabet-based Keyboard Sentences	Participant's Bliss Sentences	Notes
		<p>The participant got the last Bliss-word “lake” (×~) wrong. Even though this looks similar the Bliss symbol for multiplication should be smaller (×) and the Bliss symbol “water” (~) should in ground level.</p>

*Table 13: Participant 2 Bliss alphabet-based Keyboard Sentences*

P2 has copied the one sentence with only one minor mistake.

### 4.3.3 Participant 3 – P3

P3 is a highly competent user of an AAC system based on word prediction. They accessed the Bliss Writer system on their own iPad with touch access. They completed all the sentences using the Bliss alphabet-based keyboard. The following table is a comparison of the original Bliss sentences and participant's typed Bliss sentences.

Bliss alphabet-based Keyboard Sentences	Participant's Bliss Sentences	Notes
$\perp_3 \quad \dot{\rightarrow} \sim \quad \square \quad \times \sim$	$\perp_3 \dot{\rightarrow} \sim \square \times \sim$	The participant has not typed the spaces between Bliss-words.
$\perp_{1+} \quad \check{\delta}_1. \quad \pi \quad \zeta \pi$	$\perp_{1+} \check{\delta}_{1.3} \pi \zeta \pi$	The participant has not typed the spaces between Bliss-words and the participant has typed a Bliss number 3 between the “favourite” ( $\check{\delta}_1.$ ) and “animal” ( $\pi$ ) Bliss-words.
$\perp_3 \quad \hat{\heartsuit} + ! \quad \hat{\Delta} \rightarrow !$	$\perp_3 \heartsuit + ! \hat{\Delta} \rightarrow !$	The participant has written the action indicator on top of the exclamation mark instead of the heart in the Bliss-word “like” ( $\hat{\heartsuit} + !$ ). The participant also has not typed the spaces between Bliss-words
$\boxed{?} \quad \perp_1 \quad \hat{\downarrow} \quad \sim$	$\boxed{?} \perp_1 \hat{\downarrow} \sim$	The participant has not typed the spaces between Bliss-words.
$! \heartsuit \quad \hat{\pi} \quad \perp_{2+} \quad \pi \div$	$! \heartsuit \hat{\pi} \perp_{2+} \pi \div$	The participant has not typed the spaces between Bliss-words.
$\boxed{?} \quad \perp_2 \quad \hat{\heartsuit} + ! \quad \otimes \emptyset$	$\boxed{?} \perp_2 \hat{\heartsuit} + ! \otimes \emptyset$	The participant has not typed the spaces between Bliss-words.
$\perp_{1+} \quad \hat{\Delta} \quad \hat{\boxminus} \quad \perp_1 \quad \mathbb{D}_4$	$\perp_{1+} \hat{\Delta} \hat{\boxminus} \perp_1 \mathbb{D}_4$	The participant has not typed the spaces between Bliss-words and the participant has typed a Bliss number 6 between “me” ( $\perp_1$ ) and “April” ( $\mathbb{D}_4$ ) Bliss-words.

Table 14: Participant 3 Bliss alphabet-based Keyboard Sentences

P3 has copied all the correct Bliss-characters in all the sentences with three minor mistakes. An interesting fact about P3’s written Bliss sentences is that they did not type the space character in any of the sentences. However, this may be due to the fact that as the participant’s communication system uses word prediction. They might not be used to typing the space as the word prediction adds a space when they select a predicted word.

### 4.3.4 Participant 4 – P4

P4 was the only participant whose access method was eye-gaze. Since the Bliss Writer system is not designed to access via an eye-gaze, P4 had to rely on their communication partner. P4’s communication partner would go through each option in the Bliss Writer system and then P4 would let the communication partner know the option they want to select. P4 completed five sentences from the Bliss alphabet-based keyboard sentences. The following table is a comparison of the original Bliss sentences and P4’s typed Bliss sentences.

Bliss alphabet-based Keyboard Sentences	Participant’s Bliss Sentences	Notes
l <sub>3</sub> þ~ □ x~	l <sub>3</sub> þ~ □ X~	The participant got the last Bliss-word “lake” (x~) wrong. Even though this looks similar to the Bliss-character for “much” (x) should be smaller.
l <sub>1+</sub> ð <sub>1.</sub> æ cæ	l <sub>1+</sub> ð <sub>1.</sub> æ cæ	The participant has not made any mistakes in this sentence. The only difference is that there are two spaces between the Bliss-words “My” (l <sub>1+</sub> ) and “favourite” (ð <sub>1.</sub> ). This could be a mistake from the communication partner.
l <sub>3</sub> ♥+! △→!	l <sub>3</sub> ♥+! △→!	The participant has used a wrong character to write the Bliss-word “like” (♥+!). The participant has used the bigger “plus” (+) Bliss-character instead of small “plus” (+) Bliss-character.
l <sub>?</sub> l <sub>1</sub> ↓ ~	l <sub>?</sub> l <sub>1</sub> ↓ ~	The participant has not made any mistakes in this sentence.
!♥ # l <sub>2+</sub> #÷	!♥ # l <sub>2+</sub> #	The participant has not completed the Bliss-word “coat” (#÷)

Table 15: Participant 4 Bliss alphabet-based Keyboard Sentences

P4 has completed five sentences out of seven with few mistakes. They copied one sentence without any mistakes, four sentences with one minor mistake on each sentence and one sentence with one missing character.

### 4.3.5 Participant 5 – P5

P5 did not write any of the sentences on either of Bliss keyboards. P5’s communication device is a dedicated AAC device with word prediction. The participant was given a laptop with a joystick to access the Bliss Writer system. Even though P5 has had some experience with a joystick as an access method, since it is not their primary access method, they did not find it comfortable to access the Bliss Writer system. Because of this the participant did not complete any of the sentences.

### 4.3.6 Participant 6 – P6

P6 uses a dedicated AAC system with touch access with a keyguard. The keyguard helps them to use the touch access more accurately. The Bliss Writer system was not designed to access with a keyguard. P6 therefore used their own joystick to access the Bliss Writer system on their own AAC device. P6 completed 2 ½ sentences using the Bliss alphabet-based keyboard and a one sentence on Bliss stroke-based keyboard. The following table is a comparison of the original Bliss sentences and P6’s typed Bliss sentences.

Bliss alphabet-based Keyboard Sentences	Participant’s Bliss Sentences	Notes
$\perp_3$ $\rightarrow \sim$ $\square$ $\times \sim$	$\perp_3 \rightarrow \sim \square \times \sim$	The participant has not typed the spaces between Bliss-words and on the word “swam” ( $\rightarrow \sim$ ) the past-tense indicator has placed on “water” ( $\sim$ ) rather than on “motion through” ( $\rightarrow$ ).
$\perp_1 +$ $\check{\text{O}}_1$ $\text{A}$ $\text{C}$	$\perp_1 +$ $\check{\text{O}}_1$ $\text{A}$ $\text{C}$	The participant has not typed the dot in the Bliss-word “favourite” ( $\check{\text{O}}_1$ ).
$\perp_3$ $\hat{\text{H}} + !$ $\hat{\text{A}} \rightarrow !$	$\perp_3 \hat{\text{H}} + !$	The participant only completed half of this sentence. The participant has not typed the space between the Bliss-

		words “he” (L3) and “like” (♡+!). The participant has also used a wrong character to write the Bliss-word “like” (♡+!). The participant has used the bigger “plus” (+) Bliss-character instead of small “plus” (+) Bliss-character.
--	--	---

Table 16: Participant 6 Bliss alphabet-based Keyboard Sentences

Bliss Stroke-based Keyboard Sentences	Participant’s Bliss Sentences	Notes
L3 ♡ } ♂ □ □	L3 ♡ } ♂ □ □	The participant had selected the wrong square Bliss-character to write the Bliss-word “buy” (♂□). The participant has used the big Square “enclosure” (□) Bliss-character instead of the small Square “thing” (□) Bliss-character.

Table 17: Participant 6 Bliss Stroke-based Keyboard Sentences

P6 is one of only two participants who used the Bliss-character and Stroke keyboards. P6 copied three sentences in the Bliss-character keyboard, one sentence with minor mistake, one sentence without any spaces, but identical and finally an uncompleted sentence with a minor mistake. On the Bliss Stroke keyboard, P6 completed a one sentence with a minor mistake.

#### 4.3.7 Participant 7 – P7

P7 was unable to attend the groups session and had a separate one-to-one session with their communication partner. P7 has fairly good Bliss knowledge and the participant used both the Bliss alphabet-based, and the Bliss stroke-based keyboards. They used an iPad with touch to access the system. The following tables shows a comparison of the Bliss sentences and P7’s typed Bliss sentences on Bliss alphabet-based and stroke-based keyboards.

Bliss alphabet-based Keyboard Sentences	Participant's Bliss Sentences	Notes
$\perp_3$ $\rightarrow \sim$ $\square$ $\times \sim$	$\perp_3$ $\rightarrow \sim$ $\square$ $\times \sim$	The participant got the last Bliss-word “lake” ( $\times \sim$ ) wrong. Even though this looks similar the Bliss-character for “much” ( $\times$ ) should be smaller.
$\perp_{1+}$ $\hat{\delta}_1$ $\pi$ $\zeta \pi$	$\perp_{1+}$ $\hat{\delta}_1$ $\pi$ $\zeta \pi$	The participant has not made any mistakes in this sentence.
$\perp_3$ $\hat{\heartsuit} + !$ $\hat{\Delta} \rightarrow !$	$\perp_3$ $\hat{\heartsuit} + !$ $\hat{\Delta} \rightarrow !$	The participant has used a wrong character to write the Bliss-word “like” ( $\hat{\heartsuit} + !$ ). The participant has used the bigger “plus” ( $+$ ) Bliss-character instead of small “plus” ( $+$ ) Bliss-character.
$\boxed{?}$ $\perp_1$ $\hat{\downarrow}$ $\sim$	$\boxed{?}$ $\perp_1$ $\hat{\downarrow}$ $\sim$	The participant has not made any mistakes in this sentence.
$! \heartsuit$ $\hat{\#}$ $\perp_{2+}$ $\hat{\#} \div$	$! \heartsuit$ $\hat{\#}$ $\perp_{2+}$ $\hat{\#} \div$	The participant has written the Bliss-word “your” ( $\perp_{2+}$ ) wrong. They have used the ground level plus Bliss symbol ( $+$ ) instead of the earth line-level plus symbol which indicates “belongs to” ( $+$ ) in Bliss.

Table 18: Participant 7 Bliss alphabet-based Keyboard Sentences

Bliss Stroke-based Keyboard Sentences	Participant's Bliss Sentences	Notes
$\Delta_3$ $\heartsuit \rangle$ $\hat{\$} \square$ $\hat{x}$	$\Delta_3$ $\heartsuit \rangle$ $\hat{\$} \square$ $\hat{x}$	The participant had selected the wrong square Bliss-character to write the Bliss-word “buy” ( $\hat{\$} \square$ ). The participant has used the small square Bliss-character in the upper level ( $\square$ ) (a meaningless character used only in lexicography) instead of the small Square ( $\square$ ) Bliss-character which indicates a “thing”.















    	    	<p>The participant has made a few minor mistakes in this sentence. They have not typed a space between the Bliss-words “question” () and “tomorrow” (). The participant also written the Bliss-word “tomorrow” () with the wrong Bliss-character parentheses. They have used the bigger parentheses (()) Bliss-character “future” instead of the smaller parentheses which indicates “later” (c)</p>
---	---	---

Table 19: Participant 7 Bliss Stroke-based Keyboard Sentences

P7 copied sentences on both the Bliss-alphabet and stroke-based keyboards. From the five sentences P7 has copied, there were two sentences without any mistakes and the rest of the sentences had minor mistakes. For the two sentences in the Bliss stroke-based keyboard the participant made one minor mistake in one of the sentences and the other had a couple of small mistakes.

## 4.4 Study 1 Analysis and Discussion

Participants made common mistakes in copying Bliss in User Study 1. One of the most common mistakes was that the participant would write the Bliss-word with almost identical Bliss-characters, but some or all of the Bliss-characters in the Bliss-word would be the wrong size or in the wrong position. For an example, how the participants have written the Bliss-word “lake” () is most interesting. Of all the six participants who wrote the Bliss-word “lake”, only one participant (P3) has written the Bliss-word correctly.

Participant	Bliss-Word “lake” (x~) attempt
P1	×~
P2	×~
P3	x~
P4	×~
P6	×~
P7	×~

*Table 20: Participants attempts on writing Bliss-word “lake”*

The participants who have written the Bliss-word “lake” incorrectly used the large “multiplication” Bliss-character (×) rather than the smaller character on the earth line-level – the “much” Bliss-character (x). A similar mistake occurred when a participant would sometimes select a Bliss-character that was identical in shape, but not identical in terms of the position and size of the Bliss-character.

The researcher believes this could be because in the Bliss Writer system when the participants are given the option to select the Bliss-characters, under the Bliss-alphabet shape “Cross” (+) on the keyboard, the smaller ground level Bliss-character “much” (x) appears later than the bigger multiplication Bliss-character (×). The other reason for selecting the incorrect size of the Bliss-character is possibly due to the fact that participant was not aware of the importance of the size or position of the Bliss-characters. The correct targeting for the Bliss-character “x” grouping can therefore be regarded as successful. A number of participants suggested that if there was some reference line relative to the Bliss-characters to identify each character’s position, it would have been easier to distinguish different Bliss-characters that look similar.

In respect to the research questions, Study 1 demonstrated that it is possible to design a keyboard input method for the Bliss-character set (RQ1). Two input methods were implemented and disabled users were able to access Bliss-characters in a copy typing



task. However, during the analysis of the participant data the researcher noticed that, in the Bliss stroke-based input method, the participants were presented a large character set when they typed the first Bliss-alphabet stroke. This displays all the Bliss-characters with that stroke. This list could be something small like four Bliss-characters or long as 295 Bliss-characters. On average the user would be presented with 88 Bliss-characters in their first typed Bliss-alphabet stroke. In the Bliss alphabet-based input method, the user is presented with 41 Bliss-characters on average for each Bliss-alphabet key press. This led the researcher to look into solving the Bliss stroke-based large character set problem.

## 5 The Bliss Stroke-based Large Character Set Problem

This chapter researcher lays out the solution to the Bliss stroke-based large character set problem.

Regarding the Bliss-character access speed using the Bliss alphabet-based and stroke-based input methods, the data from first user study shows some interesting findings. The Bliss alphabet-based input method presents Bliss-characters according to how they have been categorised according to the Bliss-alphabet shapes which are presented to the user in the keyboard. Table 21 shows how many Bliss-characters are displayed to the Bliss user under each Bliss-alphabet shape when using the Bliss alphabet-based and Bliss stroke-based input methods (in the Bliss stroke-based input method, the results are only for the first stroke). On average, 41 Bliss-characters are presented in the Bliss alphabet-based input method and 88 Bliss-characters are presented in first stroke of the Bliss stroke-based input method.

Bliss-Alphabet Shape	Number of Bliss-characters (Alphabet-based)	Number of Bliss-character (Stroke-based)	Bliss-Alphabet Shape	Number of Bliss-characters (Alphabet-based)	Number of Bliss-character (Stroke-based)
⌋ Wavy Line	43	86	□ Open Square	22	29
♡ Heart	13	13	▭ Open Rectangle	25	36
⌘ Cross Hatch	3	4	△ Right Triangles	18	22
⌠ Building	26	29	· Dot	24	91
⌒ Ear	3	6	∧ Right Angles	114	295
↑ Arrow	65	114	⊥ Lines on a Base	9	63
⊗ Wheel	26	32	⊕ Cross	25	50

○ Large Circle	51	67	△ Isosceles Triangle	16	58
◦ Small Circle	81	154	∧ Symmetric Acute Angle	53	87
⤿ Half Circle	103	193	∟ Asymmetric Acute Angle	14	21
⤿ Quarter Circle	41	95	— Horizontal Line	92	336
) Parentheses	65	94	Vertical Line	92	272
□ Square	60	85	/ Slanted Line	18	24
▭ Rectangle	13	24	↗ Diagonal Line	31	96

Table 21: Number of Bliss-Characters Displayed to the user with one Bliss-Alphabet Shape

As shown in Table 21, in the Bliss alphabet-based input method, when a user presses a Bliss-alphabet shape on the keyboard, they are presented with a number of Bliss-characters in a list. This method is ideal for a Bliss user who knows the Bliss-alphabet shape group each Bliss-character belongs to. But if the Bliss user does not know that information, they can use the Bliss stroke-based input method, as they only need to know how the Bliss-character is formed. They can access Bliss-characters by typing Bliss-alphabet strokes to filter the Bliss-characters to get the exact Bliss-character they are looking for. This method is ideal for accessing Bliss-characters that are formed with multiple Bliss-alphabet strokes such as; “lorry” (𐀀𐀁) and “curtain” (𐀀𐀁). However, the Bliss stroke-based input method is not ideal for accessing Bliss-characters with one Bliss-alphabet shape such as; “forward” (→), “mind” (⤿) and “protection” (∧). As shown in Table 21, this is because when the Bliss user types any of these Bliss-alphabet strokes in the Bliss stroke-based input method, the Bliss user is presented with all the Bliss-characters with the specific Bliss-alphabet stroke. For example if the user is to write Bliss-character “protection” (∧) using the Bliss stroke-

based input method, after selecting the “Right Angles” ( $\wedge$ ) stroke they would have to look through a list of 295 Bliss-characters to find the “protection” ( $\wedge$ ) Bliss-character. In the Bliss alphabet-based input method, they would only have to go through a list of 114 Bliss-characters.

To solve this problem the researcher implemented a third input method. This is the Bliss hybrid-based input method, a combination of Bliss alphabet-based and stroke-based input methods.

## 5.1 Bliss Hybrid-based Input Method

The Bliss hybrid-based input method is a combination of the Bliss-alphabet and Bliss stroke-based input methods. The analysis data from the two previous input methods were used to implement this Bliss hybrid-based input method. When the user selects the first Bliss-alphabet key on the Bliss-based keyboard, the Bliss Writer system displays the Bliss-character set using the Bliss alphabet-based input method. From the second Bliss-alphabet key the Bliss user selects, the system retrieves Bliss-characters using the Bliss stroke-based input method. The user interface is exactly the same as the Bliss stroke-based keyboard (Figure 15).



Figure 16: Bliss Writer – Hybrid Keyboard

## 6 Study 2

This chapter describes the Study 2 participants, methodology, results and the analysis. single participant (P1) took part in study 2. The participant was given a set of sentences to copy using the three Bliss-character input methods: Bliss alphabet-based, stroke-based and hybrid-based. Study 2 was conducted as a pilot evaluation of the hybrid-based input method. The researcher also analysed the study results to compare the writing speed of each Bliss-character input method. The researcher is aiming to answer the final research question in this study.

*RQ3. How do different keyboard input methods impact on the speed of typing Bliss?*

### 6.1 Participants

The study only had one participant. The participant was recruited through a volunteer AAC User Group. Ethical approval was obtained through the SSEN Ethics Committee (Appendix B). The participant also participated in Study 1 (P1) and was one of the two participants who had a Blissymbolics background.

Participant	Access method	Bliss Knowledge
P1	Touch	Has fairly good knowledge.

*Table 22: Study Participants, Access Method and their Bliss knowledge*

### 6.2 Method

The participant was given the information sheet and the consent form before the study began. The study was conducted as a one to one session in the university AAC research pod. The researcher explained everything in the information sheet and answered any questions the participant had. Once the participant signed the consent form the session began.

Since the participant took part in study 1, they used the same login details for the Bliss Writer system in Study 2. The participant was given a list of sentences to copy in each Bliss-character input method.

The Bliss sentences use a variety of Bliss-characters. The participants needed to use different Bliss-alphabet keys and Bliss indicators to copy these sentences. The following are the Bliss sentences that the participant was asked to copy.

1.  $\lfloor_2 \quad \bigcirc_1 \quad \lfloor_2 \quad \hat{\Delta} \quad \square \uparrow \square$  – Are you going to school tomorrow?
2.  $\lfloor_1 \quad \mathcal{I} \circ_{\oplus} \quad \lfloor_{1+} \quad \hat{\phi} \nearrow$  – I have lost my phone
3.  $\square \square \quad \square \quad \bigcirc \star \dagger$  – Library was closed on Christmas day
4.  $\lambda_3 \quad \uparrow \quad \hat{\square} \quad \lambda_3 \quad \mathbb{D}_4$  – His mother is coming to see him in April
5.  $\lfloor_2 \quad \lfloor_2 \quad \heartsuit +! \quad \otimes \emptyset$  – Do you like tea?

Once the participant logged into the system they were presented with three Bliss-character input methods: the Bliss alphabet-based keyboard which allows the user to access Bliss-characters using the Bliss alphabet-based input method, the Bliss stroke-based keyboard which allows the user to access Bliss-characters using the Bliss stroke-based input method, and the Bliss hybrid-based keyboard which allows the user to access Bliss-characters using a combination of Bliss alphabet-based and stroke-based input methods.

First the participant was asked to copy the five sentences using the Bliss alphabet-based keyboard. Once they finished copying the sentences, they were asked to copy the same sentences using the Bliss stroke-based keyboard and finally with the Bliss hybrid-based keyboard.

During the study the researcher assisted the participant with any questions they had. There were a few occasions, when the participant asked help finding specific characters, but mostly during the session the participant was able to copy the bliss sentences without any issues. The participant's communication partner did assist them with using the Bliss Writer system, in terms of pressing button in the Bliss writer system on the participant's iPad.

There were no observational data collected during the study. The Bliss Writer system logged the same data that was collected in Study 1. The speed of each sentence writing was calculated using the log data collected. For each sentence the first Bliss-alphabet stroke/shape press in the keyboard was taken as the starting time and the final Bliss-character selection of the sentence was taken as the finishing time. Then the duration was calculated by subtracting the sentence finishing time from the starting time.

## **6.3 The Bliss Writer Study 2 Results**

Study 2 focused on the comparison of speed of writing using all three Bliss-character input methods. The participant was asked to copy the same five Bliss sentences using three Bliss-character access keyboards.

1. Bliss Alphabet-based Input method
2. Bliss Stroke-based Input method
3. Bliss Hybrid-based Input method (Alphabet-based and Stroke-based input methods)

### **6.3.1 Participant 1 - P1**

P1 used an iPad to access the Bliss Writer system. P1's communication partner did help them in using the system. The communication partner would help in pressing keyboard buttons when P1 asked them to do so. As this was a one to one session the researcher also helped in finding Bliss-characters when the participant asked for help.

The Table 23 shows the Bliss sentences that P1 has typed with the Bliss Writer system using different Bliss-character input methods.

Alphabet-based Keyboard	Stroke-based Keyboard	Hybrid-based Keyboard
l <sub>1</sub> Q( l <sub>2</sub> ^ △ ↕ □	l <sub>1</sub> Q( l <sub>2</sub> ^ △ ↕ □	l <sub>1</sub> Q( l <sub>2</sub> ^ △ ↕ □
l <sub>1</sub> 1' o <sub>1</sub> l <sub>1</sub> + 0/	l <sub>1</sub> 1' o <sub>1</sub> l <sub>1</sub> + 0/	l <sub>1</sub> 1' o <sub>1</sub> l <sub>1</sub> + 0/
△ □ □ Q*†	△ □ □ Q*†	△ □ □ Q*†
λ <sub>3</sub> + △ -△ λ <sub>3</sub> D <sub>4</sub>	λ <sub>3</sub> + △ -△ λ <sub>3</sub> D <sub>4</sub>	λ <sub>3</sub> + △ -△ λ <sub>3</sub> D <sub>4</sub>
l <sub>1</sub> l <sub>2</sub> ♥+! 00	l <sub>1</sub> l <sub>2</sub> ♥+! 00	l <sub>1</sub> l <sub>2</sub> ♥+! 00

Table 23: Participant's Copied Bliss Sentences

P1 copied the Bliss sentences with a few minor mistakes. These mistakes could be due to the fact that the participant had not used Bliss for many years and they are no longer aware of the importance of the size or position of the Bliss-characters. Hence the Bliss sentences that P1 typed can be taken as successful attempts.

Table 24 shows the time that P1 took to copy each Bliss sentence using the three Bliss-character input methods.

Bliss Sentences	Time on Bliss Alphabet-based Keyboard	Time on Bliss Stroke-based Keyboard	Time on Bliss Hybrid-based Keyboard
l <sub>1</sub> Q( l <sub>2</sub> ^ △ ↕ □	7 Mins	4 Mins 43 Seconds	2 Mins 27 Seconds
l <sub>1</sub> 1' o <sub>1</sub> l <sub>1</sub> + 0/	3 Mins 56 Seconds	4 Mins 19 Seconds	1 Min 49 Seconds
△ □ □ Q*†	2 Mins 35 Seconds	2 Mins 58 Seconds	2 Mins 1 Second
λ <sub>3</sub> + △ -△ λ <sub>3</sub> D <sub>4</sub>	3 Mins 39 Seconds	3 Mins 52 Seconds	3 Mins 6 Seconds
l <sub>1</sub> l <sub>2</sub> ♥+! 00	2 Mins 3 Seconds	1 Min 52 seconds	2 Mins 4 Seconds

Table 24: Bliss-Character Input methods Comparison

## 6.4 Study 2 Analysis and Discussion

To solve the Bliss stroke-based large character set problem the researcher developed the Bliss hybrid-based input method, a combination of both the Bliss alphabet-based and stroke-based input methods. In the Bliss hybrid-based input method, when users type the first Bliss-alphabet shape/stroke, the Bliss-characters are retrieved using the



Bliss alphabet-based input method and from the second Bliss-alphabet stroke onwards the characters are retrieved using the Bliss stroke-based input method.

Study 2 was conducted to compare the speed of writing using the three Bliss-character input methods. The researcher was only able to conduct the study with one participant who also participated in the Study 1.

As explained in the section 6.2, the participant was given five Bliss sentences to copy using three Bliss-character input methods: Bliss alphabet-based, stroke-based and hybrid-based input methods. The five Bliss sentences the participant was given to copy are given below:

### The Bliss Sentences

1.  $\boxed{v}$   $\Omega_c$   $\perp_2$   $\hat{\Delta}$   $\triangleup$   $\uparrow$   $\square$  – Are you going to school tomorrow?
2.  $\perp_1$   $\uparrow \circ_{\oplus}$   $\perp_{1+}$   $\hat{\phi}$   $\swarrow$  – I have lost my phone
3.  $\triangleleft$   $\square$   $\square$   $\Omega^*$   $\dagger$  – Library is closed on Christmas day
4.  $\lambda_3$   $+$   $\uparrow$   $\hat{\triangleleft}$   $\lambda_3$   $D_4$  – His mother is coming to see him in April
5.  $\boxed{v}$   $\perp_2$   $\heartsuit$   $+$   $!$   $\otimes$   $\emptyset$  – Do you like tea?

Some of the common mistakes that occurred in the Study 1 were repeated in this user study as well. In terms of the Bliss-characters the participant chose to write, the sentences are shape wise the same; the only difference is the size or the position of the Bliss-character shape.

Table 24 shows the results of how much time was taken to write each sentence using each Bliss-character input method. By looking at the data, the Bliss hybrid-based keyboard has performed better than the other methods. Four sentences out of the five that were written using the Bliss hybrid were faster than both other keyboards.

The participant took 7 minutes to write the first sentence using the Bliss alphabet-based keyboard and 4 minutes and 43 seconds to write the same sentence using the Bliss stroke-based keyboard. This was just under half the speed. However, looking at the participant's log data, it took 1 minute and 34 seconds to write the Bliss-character

“Day” (☉) using the Bliss-character keyboard; this was because the participant made three errors when accessing the Bliss-character in Bliss-character keyboard. In the Bliss alphabet-based input method, the Bliss-character “day” is categorised under the Bliss-alphabet shape “Large Circle” (○, which means ‘sun’). When accessing this Bliss-character, the participant selected the Bliss-alphabet shape “Large Circle” (○) first and missed the Bliss-character “day” (☉). The participant then selected Bliss-alphabet shape “Small Circle” (◦) followed by the Bliss-alphabet shape “Horizontal line” (—) and finally the “Large Circle” (○). Using the Bliss Hybrid keyboard, it took only 2 minutes and 27 seconds to write the first sentence. The participant only made one minor error when typing the word “tomorrow” (☉◊), the participant typed it as ☉◊. Instead of selecting the small parenthesis, which translates to “later” (◊), the participant selected the larger parenthesis (◊).

The second sentence took the participant 3 minutes and 56 seconds, 4 minutes and 19 seconds, and 1 minute and 49 seconds, to write using the Bliss alphabet-based, stroke-based and hybrid-based input methods respectively. The Bliss hybrid-based input method was twice as fast as the Bliss alphabet-based and stroke-based input methods. The Bliss stroke-based input method took longer than the Bliss alphabet-based input method. This is because the Bliss stroke-based input method is slower when it comes to accessing Bliss-characters with single Bliss-alphabet stroke. The number of Bliss-characters presented to the user is higher having typed the first stroke using the Bliss stroke-based input method (Table 21). In the second sentence there are three Bliss-characters with single Bliss-alphabet shapes/strokes. The Bliss-character “person” (⌚) appears twice and the Bliss-character “belongs to” (+) appears once. The log data shows that when using the Bliss alphabet-based input method it took as average of 3 seconds to write the Bliss-character “person” (⌚) and 4 seconds to write the Bliss-

character “belongs to” (+) while in the Bliss stroke-based input method it took an average of 12.5 seconds to write the Bliss-character “person” (⌊) and 14 seconds to write the Bliss-character “belongs to” (+). It is clear that in typing the second sentence, the participant took more than three times the time to access these Bliss-characters.

The sentences two, three and four had similar results with the Bliss Hybrid keyboard being the fastest, the Bliss alphabet-based input method being the second fastest and the Bliss stroke-based input method being the slowest of three input methods.

The fifth and final sentence had results that are different from previous four sentences. The Bliss stroke-based input method was the fastest with 1 minute and 52 seconds. The Bliss alphabet-based input method took 2 minutes and 3 seconds, and the Bliss hybrid-based input method took 2 minutes and 4 seconds to write the sentence. The fact that the Bliss Hybrid keyboard was the slowest is interesting since it had been the fastest for all the other sentences. The log data shows that the participant had made some errors using all three Bliss input methods when copying the final sentence. Using the Bliss alphabet-based input method, six errors were made; using the Bliss stroke-based input method, two errors; and using the Bliss hybrid-based input method, nine errors were made. These error rates correspond with the time it took the participant to copy the final sentence using each input method. The Bliss hybrid-based input method having the most errors gives an explanation to why it had the slowest time on the 5<sup>th</sup> sentence among the three Bliss-character input methods.

## 7 Discussion

This research extends the current support for writing Bliss. Bliss has been used as a symbol set in AAC. This has been demonstrated in systems like BlissOnline<sup>5</sup>, Simplipedia (Tillander, 2014) and Bliss OpenOffice plug-in (Lundälva, et al., 2014). All three of these systems convert the written text into Blissymbols in graphic form. With Bliss-Talk (Hunnicut, 1984) and Bliss Dasher (Ward, et al., 2000) users can access Bliss-words. The Bliss-Talk system has the ability to change the tense of any Blissymbol and is able to translate any sentence into grammatically correct English, French or Swedish. However, the Bliss-Talk system uses a magnetic keyboard with Blissymbols as the overlay. The output is in the form of synthetic speech and printed words. The system is limited to the Bliss-words that are stored in it. In contrast, Bliss Dasher uses a demonstrator Bliss font allowing users to write novel Bliss-words.

Since the beginning Blissymbolics has been a written language (Bliss, 1965). In current AAC systems that support Bliss, the users have not had access to its more than 1400 Bliss-characters. With Michael Everson's Bliss font (Everson, 2020) and the researcher's Bliss-character input methods, the user can now access the entire Bliss-characters set.

The researcher aimed to address three research questions:

- RQ1. Is it possible to design a keyboard input method for the Bliss-character set?*
- RQ2. What is the experience of people with physical impairments when accessing Bliss-characters using different access methods?*
- RQ3. How do different keyboard input methods impact on the speed of typing Bliss?*

---

<sup>5</sup> <https://www.blissonline.se/>

## **7.1 Providing Keyboard Access to the Full Bliss-Character Set**

An onscreen Bliss keyboard has been implemented based on how Chinese characters are accessed together with the Bliss “Semantographic Typewriter” idea (Bliss, 1965). This involved extending Everson’s (2020) mapping of the Bliss-alphabet on to the QWERTY keyboard to accommodate indicators and punctuation.

Current keyboard input methods for large character sets tend to rely on users being able to use phonetic and stroke-based input methods. Users can also access characters through operating systems’ character pickers. However, these techniques are unsuitable for users who may have physical and/or learning difficulties. The challenge of this research was therefore to explore ways in which disabled users could access the full Bliss-character set.

The researcher developed the Bliss Writer system with the Bliss onscreen keyboard implemented using two Bliss-character input methods (alphabet-based and stroke-based). A study was conducted with seven disabled participants (six in a group session and one participant in a one-to-one session) using the Bliss Writer system. The participants used the two Bliss input methods to copy type sentences. Participants were given seven sentences to copy using the Bliss alphabet-based input method and another seven sentences to copy using the Bliss stroke-based input method. The Bliss Writer system collected log data for each participant. With this data the researcher was able to analyse how each participant interacted with the Bliss Writer system.

The study log data shows that six out of seven participants could copy type at least one sentence using the Bliss Writer system. Three participants used both Bliss-character input methods and out of those, two participants were able complete a sentence using both Bliss-character input methods. All participants were able to access Blissymbolics at a character level.

The participants being able to copy type Bliss-characters suggests that they would be able to access any Bliss-character in the Bliss-character set. Unlike current AAC

systems that support Bliss, the Bliss Writer system's Bliss-character input methods (alphabet-based and stroke-based) provided participants with access to the full Bliss-character set and enabled them to write Bliss using a Bliss keyboard. Two participants relied on help to physically access the keyboard – however they were able to instruct a helper to operate the keyboard.

The researcher was able therefore to answer the first research question: Is it possible to design a keyboard input method for the Bliss-character set? The Bliss alphabet-based and stroke-based input methods enable users to access any Bliss-character from the Bliss-character set. Users can write in Blissymbols using a Bliss-alphabet keyboard and the Bliss-character input methods.

## **7.2 The Experience of Disabled Users using the Bliss Writer**

Although disabled users demonstrated the ability to access Bliss-characters using the Bliss writer system, several issues arose when copy typing sentences. In study 1 only two of the seven participants had some Bliss knowledge. All the participants made errors. Most of these errors were common across participants. These common errors include typing a Bliss-character in the wrong position or size; not including the Bliss indicator; or typing the Bliss indicator on top of the wrong Bliss-character. This suggests that in order to write in Bliss, the user needs to be aware of the underlying characteristics of Blissymbolics such as the position of characters, different sizes of Bliss-characters and how the Bliss indicators are used.

To minimise some of these errors there could be some improvements made to the Bliss Writer system. A grid behind the Bliss-characters to guide the users might help with selecting the correct size of Bliss-characters and placing the Bliss-character in the correct position between the sky line and earth line.

In terms of AAC access methods, the Bliss Writer system should be able to be used by a wide range of AAC users. In study 1 participants used touch and joystick to access the Bliss writer system. A participant who uses an eye-gaze system (P4) could not use

the Bliss Writer system with their preferred AAC access method, as the Bliss Writer system was not designed to be accessed via eye-gaze. The participant had to rely on their communication partner to access the Bliss Writer system. Similarly, P6 accesses their AAC device using a keyguard. Since the Bliss Writer keyboards were not designed to be used with a keyguard, the participant had to use their joystick to access the Bliss Writer system.

The Bliss Writer system needs to be accessed by a wide range of disabled users. This includes the need to implement switch access and eye-gaze. For the users with keyguard use would benefit having a Bliss keyboard that can be customised according to their requirements.

### **7.3 The Large Character Set Problem**

The researcher has developed two Bliss-character input methods (alphabet-based and stroke-based) that provide access to the full Bliss-character set. By doing so it has been possible to explore the challenges which would be faced by disabled user in navigating through larger set of characters.

More than a thousand different implementations of Chinese input methods have been developed (Po, et al., 2009). Using these different character input methods require good dexterity, the ability to use phonetic spelling of characters and /or the ability to ‘type’ characters using a shape-based keyboard. These strategies would be difficult for users with physical and/or learning disabilities.

This research has addressed this challenge within the context of Bliss by developing a hybrid-based input method, a combination of Bliss alphabet-based and stroke-based input methods. This addressed the “Bliss large character set problem” – when using the Bliss stroke-based input method, users are presented with an unwieldy number of Bliss-characters when these have only a single Bliss-alphabet shape. Accessing single shape characters using the Bliss alphabet-based input method presents smaller numbers of characters. This led to the hybrid-based input method. Having a similar user interface (UI) to the Bliss stroke-based input method, the hybrid-based input method uses the Bliss alphabet-based input method to retrieve Bliss-characters in the

first Bliss keyboard shape/stroke key press from the user. From the second Bliss keyboard stroke key press the Bliss-characters are retrieved using the Bliss stroke-based input method.

The researcher conducted a study with a one participant as a pilot evaluation of the Bliss hybrid-based input method and to compare the writing speed of each Bliss-character input methods. The participant was given five Bliss sentences to copy in all three Bliss-character input methods. The Bliss Writer system collected log data in each interaction of the participant. The researcher analysed the log data and compared the writing speed of each Bliss-character input method.

Results of Study 2 show that Bliss hybrid-input method performed well in terms of writing speed, it was the fastest in four out of five sentences. Bliss alphabet-based input method was 2<sup>nd</sup> fastest in four sentences and 3<sup>rd</sup> fastest in 1 sentence. The stroke-based input method was the fastest in one sentence, 2<sup>nd</sup> fastest in one sentence and 3<sup>rd</sup> fastest in three sentences.

## **7.4 Limitations of the Research**

There were two main limitations in the research; how study 1 was conducted and the number of participants in the two studies. The researcher conducted study 1 as a group session. The session started with a Bliss introductory training session that was conducted by Professor Annalu Waller. Then the participants were given their login details to the Bliss Writer system along with sentences they needed to copy with each Bliss-character input method. The researcher and his colleagues assisted the participants with any queries they had. However, the researcher believes the participants would have had a better experience if the study had been a one-to-one session. During the group session, even though the researcher had assistance from his colleagues, the researcher found the session to be a bit chaotic. The session with P7 was a one-to-one session and researcher believes it was a better experience for both the participant and the researcher. The researcher was able to give his full attention to the participant during the session and was able answer any questions or assist with any difficulties promptly.



If the researcher was to do this study again, he would do the study in two stages.

1. The Blissymbol introductory training session that was conducted by Professor Annalu Waller as a group session, followed by an introductory demonstration of the Bliss Writer.
2. The Bliss sentence writing on the Bliss Writer system would be conducted in individual one to one sessions.

This would make the study a better experience for the participants and the researcher would be able to give his full attention to each participant during their session.

The small number of participants was due to the difficulty of finding disabled Bliss users. In study 1 only two of the participants (P1 and P7) had any Bliss background and in Study 2, the participant had Bliss background (P1 from study 1). Having seven participants in Study 1 gave a good indication of the ease of access to the full Bliss-character set on different input methods and the experience of people with physical impairments when accessing Bliss-character set. However if there had been more participants, it would have painted a broader picture of any limitations in Bliss-character input methods and the experiences of disabled users when accessing Bliss-characters.

In Study 2 the researcher only had one participant who had some Bliss knowledge. In an ideal situation the researcher would have liked to have had more participants in Study 2, however due to the time constraint the study was conducted with only one participant.

## **7.5 Future Work**

This section outlines some of the future work that the researcher would plan to do with the Bliss Writer system and Bliss-character input methods.

## 7.5.1 Implementing further Bliss-Character input methods

In Chinese there have been more than 1000 different implementations of input methods (Po, et al., 2009). Having implemented three Bliss-character input methods, the researcher believes implementing more Bliss-character input methods would benefit the users in terms of choosing the best suitable input method for their need. In the Blissymbol Reference Guide (Wood, et al., 1992), the Bliss-characters are categorised according to the Bliss-alphabet shapes. Each Bliss-character is categorised under a Bliss-alphabet Shape. Each Bliss-alphabet shape can have different variations within Bliss-characters. These variations can be in shape (# #), size (□ □), orientation (↑ → ↓ ←) or vertical position (° ° ° °). Each Bliss-alphabet shape can have its own different variations. Table 25 shows some of the Bliss-characters and their variations according to the Blissymbol Reference Guide (Wood, et al., 1992).

}	{ } ~ ~ ~ }
♡	♡
#	# #
⤿	⤿ ⤿ ⤿ ⤿ ⤿ ⤿ ⤿ ⤿ ⤿ ⤿
□	□ □ □ □ □ ◇

Table 25: Variations of Bliss-Alphabet Shapes

The work that has done in the Bliss stroke-based input method would be extended to implement a new Bliss-character input method to allow for access using the Bliss-alphabet shape variations. The user would type a Bliss-alphabet shape on the Bliss keyboard and they would then be presented with different variations of that Bliss-alphabet shape in order to select the relevant Bliss-alphabet shape variation. The user will then be presented with all the Bliss-characters derived from the chosen Bliss-alphabet shape variation. This would utilise the analysis of all the Bliss-characters by the researcher as described in section 3.2.2. The researcher believes this would allow Bliss users to access Bliss-characters faster and more efficiently.

## **7.5.2 Natural Language Processing in Blissymbolics**

In terms of Natural Language Processing (NLP) in Blissymbolics, NLP techniques have not been used to enhance Bliss use with the exception of systems like Bliss-Talk (Hunnicutt, 1984) and Bliss-Dasher (MacKay, et al., May 2007). However, the work that has been done by Lundälv et al. (2014) relating to multi-modal and multilingual language support for a wide area of applications is a steppingstone for the use of NLP in Bliss. The Concept Coding Framework (CCF) has been developed: this is an open-source framework technology infrastructure for making multilingual and multi-modal language resources available in mainstream Information and Communications Technology (ICT) environments as well as in Assistive Technology (AT). Part of CCF is the “CCF-SymbolServer” (Mats, et al., 2011). This is a graphical symbol server; currently it supports two symbol systems, Blissymbolics and ARASAAC<sup>6</sup> and four languages (English, Swedish, Spanish, and Dutch). Any word from a given language can be sent to the CCF-SymbolServer, which then looks up meanings or concepts in its databases and returns symbol representations from the relevant symbol systems. The researcher would like to implement a system for Bliss-word prediction using these approaches. This would involve evaluating the efficiency of Bliss writing when implementing prediction.

## **7.5.3 Further Development on Bliss Writer System**

The researcher intends to further develop the Bliss Writer system into a Bliss-word processor with more accessibility using Bliss-character input methods. The researcher would also like to develop the Bliss Writer system with word processor capabilities such as; copy, past, save, export to PDF, share and print. Having these word processor functionalities would enable users to write Bliss in a rich text environment, they would

---

<sup>6</sup> <http://www.arasaac.org/>

also be able to share Bliss documents with others and publish documents in Blissymbols.

## 8 Conclusion

Communication is an essential part of what makes us human. It enables us to initiate and maintain daily interactions with each other. Most people do not think about the mechanics required to communicate; it is intuitive. However, there are many people who require Augmentative and Alternative Communication (AAC) systems to communicate, due to speech and/or cognitive impairments.

High-tech AAC systems provide users with access to meaning with written words, symbols, and images. An AAC user uses the AAC system to communicate using written words (pre-stored or composed by the user with the help of word prediction) or symbol sets such as Picture Communication Symbols (PCS) and Widget (Rebus Symbols). Using symbol sets means AAC users are limited to the symbols available in the available set and this leads to challenges expressing anything outside the symbol set. Blissymbolics, however, is an ideographic written language that can respond to this issue. Blissymbolics allows the AAC user to communicate with anyone using Bliss-characters. Even though there are more than 1400 Bliss-characters the Bliss user can use these characters to create a potentially unlimited number of Bliss-words with conceptual meaning, in order to communicate. A difficulty with Blissymbolics is that there is as yet no way to give Bliss users access to the full Bliss-character set in such a way as to generate Bliss-words dynamically. Because of this Bliss has been used as a symbol set rather than a written language.

This thesis has focused on developing an efficient Bliss-character input method so that Bliss users can write in Bliss rather than use it as a symbol set. In section 2.6 the researcher gave an overview of some of the technologies that have been developed using Blissymbolics. These technologies have many limitations for Bliss users in terms of using Blissymbolics, due to the fact that Bliss has mainly been used as a symbol set. Bliss users have never yet been given the access to the whole set of Bliss-characters to write in Bliss.

In this thesis the researcher aimed to answer following three research questions.

- RQ1. Is it possible to design a keyboard input method for the Bliss-character set?*
- RQ2. What is the experience of people with physical impairments when accessing Bliss-characters using different access methods?*
- RQ3. How do different keyboard input methods impact on the speed of typing Bliss?*

To address these research questions the researcher developed a Bliss Writer system with three Bliss-character input methods; Bliss alphabet-based, stroke-based, and hybrid-based input methods which allow Bliss users to write in Bliss using the full Bliss-character set.

The Bliss alphabet-based input method allows Bliss users to access Bliss-characters using each Bliss-alphabet shape by which they (Bliss-characters) are categorised. The Bliss alphabet-based input method requires the Bliss user to have some knowledge of which Bliss-alphabet shape category to which each Bliss-character belongs.

The Bliss stroke-based input method allows the Bliss user to access Bliss-characters using the strokes (elemental shapes used in Bliss) from which each character is formed. The researcher made an analysis of the Bliss-alphabet strokes which are used to form each Bliss-character. This allows Bliss users to access Bliss-character by typing the Bliss-alphabet shapes as strokes.

*The Bliss hybrid-based input method is a combination of the Bliss alphabet-based and stroke-based input methods. This input method was implemented to solve the “Large Character Set” problem in the Bliss stroke-based input method where the Bliss user is presented with a large Bliss-character set after the first stroke is typed. Chapter 5 describes the problem and Table 21*  
*Table 21: Number of Bliss-Characters Displayed to the user with one Bliss-Alphabet Shape*

shows the comparison of number of Bliss-characters presented to the Bliss user using the Bliss alphabet-based and stroke-base input methods. The Bliss hybrid-based input

method allows Bliss-characters to be accessed using both Bliss alphabet-based and stroke-based input methods; when the user types the first Bliss-alphabet shape the system uses the Bliss alphabet-based input method to retrieve Bliss-characters after which Bliss-characters are retrieved using the Bliss stroke-based input method.

In study 1 the Bliss-character input methods were evaluated with seven participants who were given five sentences to copy-type using Bliss alphabet-based and stroke-based input methods on the Bliss Writer system. Five participants used touch access and two participants used a joystick to access the Bliss Writer system. One of the participants who uses eye-gaze was not able to use the Bliss Writer system in that way as it had not been designed for access via an eye-gaze method. That participant's communication partner accessed the system via touch following the directions given by the participant. The other participant who used the joystick to access the Bliss Writer, usually accesses their AAC system with direct selection using a keyguard. The Bliss Writer system had not been designed to be used with the participant's regular keyguard. So that participant accessed the Bliss Writer system using a joystick. Out of seven participants, six were able to complete at least one sentence. Four participants completed at least five sentences on Bliss alphabet-based input method and two participants copy-typed sentences using Bliss alphabet-based and stroke-based input methods.

The researcher conducted study 2 with one participant to evaluate the Bliss hybrid-based input method and analysed the study results to compare the writing speed of all three Bliss-character input methods. The participant copy typed the same five sentences using all three Bliss-character input methods. After the researcher analysed the sentence completion times on each Bliss-character input method, it was found that the hybrid-based input method was the fastest on four sentences.

By developing these three Bliss-character input methods the researcher was able answer the RQ1: Yes, it is possible to design a keyboard input method for the Bliss-character set.

Even though Bliss-character input methods give access to the full Bliss-character set, in both user studies the participants made some similar errors when accessing Bliss-

characters. Since different sizes and position of the Bliss-character shape between the earth and sky line can give different meaning to Bliss-words, it is important that when writing Bliss the users are able to select the correct form of the Bliss-character. As described in section 4.4, in study 1 when writing the word “Lake” (x~) participants has taken different approaches (Table 20). Only one participant (P3) used the correct combinations of the Bliss-characters. Other participants used similar Bliss-characters with different sizes or positions between earthline and skyline. One of the reasons for selecting the incorrect size or positioned Bliss-characters is possibly due to the fact that participants are not aware of the importance of the size or position of the Bliss-character. Another might be a limitation in the display design, which could in principle contain both skyline and earthline for orientation. The researcher believes that having a grid behind where Bliss-characters are displayed would help with selecting the correctly sized and positioned Bliss-characters.

Having implemented three Bliss-character input methods in the Bliss Writer system, it is important that wide range of disabled users can access this system. Currently the Bliss Writer system can only be accessed with direct access methods such as touch and joystick access. Implementing different accessibility methods such as switch access and eye-gaze to access the systems would give a wider range of disabled users a better experience on accessing the Bliss Writer system as they would be able to use their primary AAC access method to use the Bliss Writer system.

In terms of the user experience the research identified that in study 1 participants struggled to differentiate Bliss-characters with their different sizes and positions. The researcher suggests improving the UI of Bliss Writer system so the participants can differentiate different characters with sizes and position. Also implementing AAC access methods so that a wide range of disabled users can write Bliss. The researcher believes there needs to be further evaluations of the Bliss Writer system and Bliss-character input methods to understand the most significant issues people with physical impairments have when accessing Bliss-characters using different AAC access methods.



In the study 2 of the research, as well as evaluating the Bliss hybrid-based input method, the researcher was aiming to address the RQ3: How do different Bliss-character input methods impact the speed of typing Bliss? The study was conducted with one participant. The participant was given five sentences to copy-type using the three Bliss-character input methods. The participant's interactions with the Bliss Writer system was logged and the data was analysed to calculate the duration it took to write each sentence using the three Bliss character input methods. Chapter 6 discusses the results from the user study 2 where, when using the Bliss hybrid-based input method, the participant was able to copy type four sentences (out of 5) faster than the Bliss alphabet-based and stroke-based input methods. However as described in the section 7.4; "Limitations of the Research", the study only had one participant. This has given the researcher reason to believe that these Bliss-character input methods need to be evaluated with a wider group of disabled users with wide range AAC access methods. This would enable the researcher to analyse the writing speed of each Bliss-character input method on different AAC access methods as well as an overall Bliss writing speed across different AAC access methods.

The researcher believes that the research hypothesis has shown that it is possible to design and implement an input method system that allows disabled Bliss users to generate Bliss-words by writing using Blissymbolics. The basic analysis of Bliss-characters by alphabet-class and by stroke composition is sound, and can be refined in future work so that when Bliss finally gets encoded in Unicode these input methods, and particularly the hybrid system, can be refined as Bliss-character input methods. The researcher hopes to develop a Bliss Writer word processor with different AAC access methods, so the users can write and share documents written in Bliss.

## 9 References

Beukelman, D. & Mirenda, P., 2014. *Augmentative & Alternative Communication: Supporting Children and Adults with Complex Communication Needs*. Fourth Edition ed. s.l.:Paul H. Brookes Publishing.

Bliss, C. K., 1965. *Semantography (Blissymbolics)*. Second enlarged Edition ed. s.l.:s.n.

Blissymbolics Communication International, 2009. *The fundamental rules of Blissymbolics: creating new Blissymbolics characters and vocabulary*, s.l.: s.n.

Bluche, T. & Messina, R., 2016. *Faster Segmentation-Free Handwritten Chinese Text Recognition with Character Decompositions*. Shenzhen, 15th International Conference on Frontiers in Handwriting Recognition (ICFHR).

Borgestig, M. et al., 2016. Eye gaze performance for children with severe physical impairments using gaze-based assistive technology—A longitudinal study. *Assistive Technology*, 28(2), pp. 93-102.

Chen, Z. & Lee, K.-F., 2000. New Statistical Approach to Chinese Pinyin Input. *Proceedings of the 38th annual meeting on association for computational linguistics*, pp. 241-247.

Communication Matters, 2017. *Symbols*. [Online]  
Available at: <https://www.communicationmatters.org.uk/page/symbols>

Communication Matters, n.d. *What is AAC?*. [Online]  
Available at: <http://www.communicationmatters.org.uk/page/what-is-aac>  
[Accessed 14 April 2017].

European Commission's 7th FP, 2008. *AEGIS – Accessibility Everywhere: Groundwork, Infrastructure, Standards*. [Online]  
Available at: <http://www.aegis-project.eu/>

Everson, M., 1998. *Encoding Blissymbolics in Plane 1 of the UCS*. [Online]  
Available at: <http://std.dkuug.dk/jtc1/sc2/wg2/docs/n1866.pdf>

Everson, M., 2020. *Unicode*. [Online]  
Available at: <https://www.unicode.org/L2/L2020/20271-n5149-blissymbols-kbd.pdf>

Hawking, S., n.d. *My Computer*. [Online]  
Available at: <http://www.hawking.org.uk/the-computer.html>  
[Accessed 2018].

Higginbotham, D. J., Leshner, G. W., Moulton, B. J. & Roark, B., 2012. The Application of Natural Language Processing to Augmentative and Alternative Communication. *Assistive Technology*, 24(1), pp. 14-24.

Hunnicut, S., 1984. Bliss symbol-to-speech conversion: Bliss-talk. *STL-QPSR*, Volume 25, pp. 058-077.

Ichbiah, J. D., 1995. *Method for designing an ergonomic one-finger keyboard and apparatus therefor*. USA, Patent No. US5487616A.

Isaacson, M. D. & Lloyd, L. L., 2015. potential for developing a tactile communication system based on Blissymbolics. *Developmental Neurorehabilitation*, 18(1), pp. 47-58.

ISO, 1993. *ISO-IR 169*. [Online]  
Available at: <https://www.itscj.ipsj.or.jp/iso-ir/169.pdf>

Koester, H. H. & Simpson, R. C., 2017. Effectiveness and usability of Scanning Wizard software: a tool for enhancing switch scanning. *Disability and Rehabilitation: Assistive Technology*.

Leshner, G. W., Moulton, B. J. & Higginbotham, D. J., 1999. *Effects of ngram order and training text size on word prediction*. s.l., Proceedings of the RESNA'99 Annual Conference, pp. 52-54.

Light, J., 1989. Toward a definition of communicative competence for individuals using augmentative and alternative communication systems. *Augmentative and Alternative Communication*, pp. 137-144.

Lundälva, M. et al., 2014. Inclusive AAC: Multi-modal and multilingual language support for all. *Technology and Disability*, Volume 26, p. 93–103.

Lundälv, M. & Derbring, S., 2012. *Towards general cross-platform CCF based multi-modal language support*. Linz, Austria, s.n.

MacKay, D., Wills, S. & Waller, A., May 2007. *Bliss-Dasher – Efficient semantic writing with any muscle*, s.l.: s.n.

Mats, L. et al., 2011. *Graphic Symbol Support in Open/LibreOffice Shaping Up – Graphic Symbol Server and Inline Symbol Font Display Based on the CCF*. s.l., 2nd International AEGIS Conference and Final Workshop 2011.

McNaughton, D. & Light, J., 2013. The iPad and Mobile Technology Revolution: Benefits and Challenges for Individuals who require Augmentative and Alternative Communication. *Augmentative and Alternative Communication*, 29(2), p. 107–116.

McNaughton, S. H., 1998. *Reading Acquisition of Adults With Severe Congenital Speech and Physical Impairments: Theoretical Infrastructure, Empirical Investigation, Educational Application*, s.l.: s.n.

Meteor, 2012. *Creating your first app*. [Online] Available at: <https://www.meteor.com/try> [Accessed October 2014].

Meteor, 2019. *Meteor - Showcase*. [Online] Available at: <https://www.meteor.com/showcase>

MongoDB, 2019. *MongoDB - Customers*. [Online] Available at: <https://www.mongodb.com/who-uses-mongodb>

MongoDB, 2019. *The MongoDB 4.2 Manual*. [Online] Available at: <http://docs.mongodb.org/manual/> [Accessed February 2015].

MongoDB, 2019. *What Is MongoDB?*. [Online] Available at: <https://www.mongodb.com/what-is-mongodb>

Newell, A., Langer, S. & Hickey, M., 1997. The role of natural language processing in alternative and augmentative communication. *Natural Language Engineering*, 4(1), pp. 1-16.

Okrent, A., 2009. *In the Land of Invented Languages*. s.l.:s.n.

Po, L.-M. et al., 2009. *Six-Digit Stroke-based Chinese Input Method*. San Antonio, TX, USA, IEEE International Conference on Systems, Man, and Cybernetics.

Scottish Government, 2012. *A Right to Speak Supporting Individuals who use Augmentative and Alternative Communication*, s.l.: Scottish Government.

Siegel, L., 2002. The Argument for a Constitutional Right to Communication and Language. *The Journal of Deaf Studies and Deaf Education*, 7(3), p. 258–266.

Tillander, M., 2014. *Simplipedia - Making Wikipedia accessible*, Göteborg, Sweden: s.n.

Trnka, K. et al., 2007. *The effects of word prediction on communication rate for AAC*. s.l., Association for Computational Linguistics, pp. 173-176.

Ulrika, F., Britt, A. & Gunilla, T., 2009. Development and Evaluation of a Minspeak Application Using Blissymbols: Experiences from Two Case Studies. *Augmentative and Alternative Communication*, 17(4), pp. 233-244.

Várady, G., Szalmas, A. & Sik-Lányi, C., 2002. *Electronic usage of BLISS symbols*. Veszprém, Hungary, Disability, Virtual Reality and Associated Technologies.

Ward, D. J., Blackwell, A. F. & MacKay, D. J. C., 2000. *Dasher - a Data Entry Interface Using Continuous Gestures and Language Models*. s.l., ACM, pp. 129-137.

Wilkins, J. & Ratajczak, A., 2009. Developing Students' Literacy Skills Using HighTech SpeechGenerating Augmentative and Alternative Communication Devices. *Intervention in School and Clinic*, 44(3), p. 167–172.

Wood, C., Storr, J. & Reich, P., 1992. *Blissymbol Reference Guide*. Toronto: Blissymbolics Communication International.

Zheng, Y., Li, C. & Sun, M., 2011. *CHIME: An Efficient Error-Tolerant Chinese Pinyin Input Method*. Barcelona, International Joint Conferences on Artificial Intelligence.

# Appendix A.

## **Bliss Writer System login**

The researcher has made available the Bliss Writer System for anyone who would like to access. Following is the logging details and once the user login you will be presented with the three Bliss-character input methods.

Bliss Writer Link: <http://167.71.141.231/>

- Username: Guest
- Password: pwd.guest.2020

Please contact the researcher ([p.g.r.h.nandadasa@dundee.ac.uk](mailto:p.g.r.h.nandadasa@dundee.ac.uk)) if there is any issue.

## **Log files from the User Studies**

The log files from the two user studies the researcher conducted are available in a university OneDrive online storage application.

Please Click here: [User Study Log Files](#)

# Appendix B. Ethics Approval From

University  
of Dundee



School of Science and Engineering Research Ethics Committee

University of Dundee  
Dundee  
DD1 4HN

26 April 2019

Dear Hasith

Application Number: UOD-SSEREC-DoC-RPG-2019-008

Title of Project: Bliss Writing System

I am writing to advise you that your ethics application has been reviewed and approved on behalf of the School of Science and Engineering Research Ethics Committee (SSEREC).

Any changes to the approved documentation (e.g., study protocol, information sheet, consent form) must be approved by this SREC before the changes are implemented.

Requests for amendments should be requested using the [Post-Approval Request for an Amendment form](#)

Approval is valid for the duration of the project, as stated in the original application. Should you wish your study to continue beyond the stated project end date, you must request an extension to this approval a minimum of 3 months before the project end date using the [Post-Approval Request for an Extension form](#)

Yours sincerely

Dr Caroline Erolin  
Convener, School of Science and Engineering Research Ethics Committee



# Appendix C. Study 1 Questionnaire

## Questionnaire Page 1



### Bliss Writing System QUESTIONNAIRE

1. What physical access method did you use to interact with the system?

Touch	<input type="radio"/>
Joystick	<input type="radio"/>
Eye-Gaze	<input type="radio"/>

2. Ease of use of the Bliss Keyboard layout:

a. What are the positive things about the Keyboard?

b. What do you think can be improved?

3. Bliss Shape-based access method – First keyboard participants tried

a. What was the best thing about selecting the Bliss-Characters?

b. What was difficult about selecting the Bliss-Characters?

## Questionnaire Page 2



### Bliss Writing System QUESTIONNAIRE

c. What do you think can be improved?

4. Bliss Stroke-based access method - Second keyboard participants tried

a. What was the best thing about selecting the Bliss-Characters?

b. What was difficult about selecting the Bliss-Characters?

c. What do you think can be improved?

5. Which method did you prefer?

Method 1: Bliss Character Based Input Method	<input type="checkbox"/>
Method 2: Bliss Stroke Based Input Method	<input type="checkbox"/>
Neither	<input type="checkbox"/>

6. Do you have any other feedback on the Bliss Writer System?

# Appendix D. Bliss-Writer Development

## Tech Stack

### Meteor Web Framework

Meteor is a full-stack JavaScript web development framework that written in Node.js. Meteor allows rapid prototyping which allows developers to create a small web application with as little code as possible. But Meteor is also capable of developing commercial level web applications (Meteor, 2019). Meteor integrates with MongoDB (MongoDB, 2019). Meteor uses a Publish-Subscribe pattern, so when data changes in the database the client side of the website can see the changes in real-time.

### MongoDB

The database that works in Meteor is MongoDB (MongoDB, 2019). MongoDB is a NoSQL, JSON like (MongoDB calls the format BSON) document-oriented database. The schema of the database is dynamic. MongoDB is one of the most popular NoSQL databases, and it has been used by many commercial and government organisations (MongoDB, 2019). In relational databases the data is stored in tables, in MongoDB the documents are stored in collections. Each document schema in a collection can be change from each other in a collection.

When a new Meteor project is created it has its own MongoDB database, there is no setup or configuration required. So, whenever the Meteor sever runs, so does the database.

### *MongoDB vs. Relational Databases*

Since MongoDB is a document database, some of the terms that users in MongoDB is different from the relational databases. Table 26 shows the terms that used in relational databases and how they translate into MongoDB, a non-relational Database.

Relational Database	MongoDB
Database	Database
Table	Collection
Row	Document
Column	Field
Primary Key	Primary Key

*Table 26: MongoDB and Relational Database Comparison*

In Meteor, creating a collection involves calling *CollectionName = new Mongo.Collection("collection");* in the JavaScript file. This sets up a MongoDB collection called “collection” in the server, and on the client side this creates a cache connection to the database collection. In the JavaScript files this collection can be accessed by calling the “CollectionName”.

## Implementation

### Application Structure

When creating a Meteor, the project creates three files: a JavaScript file, a HTML file, and a CSS file (Meteor, 2012). By default, the JavaScript files in the project are bundled and sent to the client and server sides. The file names and folder names in the project can affect the order of how they are loaded in the Meteor project. For example, if there are two folders called “client” and “server” Meteor sends the JavaScript files in those folders according to the folder name. “Server” JavaScript files handles in the server side and the “client” JavaScript files handles in the client side. There is also “private”, “public” and “test” folders, “private” is for files that can only be accessible from the server code, “public” folder is for file like icon, public text files and files in “test” folder does not load anywhere it is for local test code. In this way the developer does not divide the code in to sever and client side in the development. Instead, code is divided within the folder and Meteor takes care of the rest. The Figure 17 shows an example of a Meteor Web application structure.

```

1  imports/
2    startup/
3      client/
4        index.js          # import client startup through a single index entry point
5        routes.js         # set up all routes in the app
6        useraccounts-configuration.js # configure login templates
7      server/
8        fixtures.js       # fill the DB with example data on startup
9        index.js          # import server startup through a single index entry point
10
11   api/
12     lists/               # a unit of domain logic
13       server/
14         publications.js   # all list-related publications
15         publications.tests.js # tests for the list publications
16         lists.js          # definition of the Lists collection
17         lists.tests.js    # tests for the behavior of that collection
18         methods.js        # methods related to lists
19         methods.tests.js  # tests for those methods
20
21   ui/
22     components/          # all reusable components in the application
23                           # can be split by domain if there are many
24     layouts/             # wrapper components for behaviour and visuals
25     pages/               # entry points for rendering used by the router
26
27   client/
28     main.js              # client entry point, imports all client code
29
30   server/
31     main.js              # server entry point, imports all server code

```

*Figure 17: Example Project Structure*

The Bliss Writer web application has a similar structure to the example project structure in Figure 17. The Bliss Writer web application contains the top-level folder: “imports”, “client” and “server”.

The “imports” folder contains all the main application code. There are three sub folders in the “imports”: “api”, “startup” and “ui”. The “api” folder contains all the application database functionality code. In the Bliss Writer application there are three MongoDB Collections: “bliss\_chars”, “bliss\_alphabet” and “bliss\_logs”. All the code for these MongoDB collections is in the “api” folder. The “startup” folder contains all code that runs when the web application starts. This includes the code that populates the MongoDB database and the code to set up the web URL routes. The code that handles the user interface is in the “ui” folder. There are three sub folders in the “ui” folder: “javascript”, “stylesheet” and “views”. The “javascript” folder contains all the JavaScript code files that handle the Meteor Template files. The “stylesheet” folder contains the CSS files and finally the “views” folder contains all the HTML files of the web application.

The top level “client” folder is the client entry point of the Meteor web application. All the client-side code from the “import” folder is imported into the code in the “client” folder.

The top level “server” folder is the sever side entry point of the Meteor web application. All the server-side code from the “import” folder is imported into the “server” folder.

- Bliss Writer Project – Meteor
  - Imports
    - api ---- contains the MongoDB Collections
      - bliss\_alphabet.js ---- Bliss-alphabet MongoDB collection operations
      - bliss\_chars.js ---- Bliss-characters MongoDB collection operations
    - startup ---- Contains the server and client startup operations
      - client
        - routers.js ---- Set up all the web URL routes in the application
      - server
        - populate\_data.js ---- Populate the MongoDB with Bliss Data.
  - ui
    - javascript
      - bliss\_chracter\_keyboard.js ---- Handles the functionalities of Bliss Character keyboard Template
      - bliss\_stroke\_keybaord.js ---- Handles the functionalities of Bliss Stroke keyboard Template
      - bliss\_writer.js ---- Handles the functionalities of Bliss Writer page Template
      - choose\_KB.js ---- Handles the functionalities of Choose keyboard page Template
      - login\_page.js ---- Handles the functionalities of participant login page Template
      - register\_page.js ---- Handles the functionalities of participant register page Template
    - stylesheets
      - bliss\_writer.css ---- Bliss Writer CSS file
    - views
      - bliss\_chracter\_keyboard.html ---- Bliss Character keyboard design template
      - bliss\_stroke\_keybaord.html ---- Bliss Stroke keyboard design template
      - bliss\_writer.html ---- Bliss Writer page design template
      - choose\_KB.html ---- Choose Bliss Keyboard page design template

- login\_page.html ---- Participant login page design template
- register\_page.html ---- Participant register page design template
- client
  - main.js
  - main.html
  - main.scss
- server
  - main.js
- private ---- contains Bliss Data files to generate the Bliss-character and Alphabet MongoDB Collections.
- public
  - fonts ---- Contains the Bliss Unicode Font files

The “private” folder contains all the Bliss-character data that is used to populate the Bliss Writer system database. The “public” folders contain the Bliss Unicode font files that are used in the Bliss Writer system.

## Database Structure

The Bliss Writer database is a MongoDB document database. The Bliss Writer system database contains three Collections: Bliss-characters Collection, Bliss-alphabet Collection and Bliss Logs Collection. The Bliss-characters collection contains all the Bliss-characters, which contains 1236 Bliss-characters. The Figure 18 below shows the schema of one of the Bliss-characters in the Bliss-characters Collection.

```
{
  "_id": "8jr7X8kX3W8f8FgQ6",
  "get_id": 23,
  "bliss_av": "24940",
  "bliss_unicode_number": "16216",
  "bliss_unicode_value": "~:",
  "bliss_unicode_name": "spray",
  "bliss_character_english": "spray,vaporization",
  "english_one": "spray",
  "explanation": "(water,fluid,liquid + a cluster of seven dots)",
  "bliss_character_based_letter": "Wavy lines",
  "bliss_stroke_based_letters": {
    "strokes": ["?", ".", ".", ".", ".", ".", ".", "."],
    "stroke_counts": [{
      "strokes": "?",
      "count": 1
    },
    {
      "strokes": ".",
      "count": 6
    }
  ]
}
```

Figure 18: Bliss-Characters DB Collection Schema

The “bliss\_character\_based\_letter” value is used retrieved the Bliss-character in the Bliss alphabet-based input method and the “bliss\_stroke\_based\_letter” value is used to access the Bliss-character in the Bliss stroke-based input method.

The Bliss-alphabet Collection contains all the Bliss-alphabet shape characters. This collection is used to create the Bliss Keyboard, the Bliss user is used to type. There are 33 documents in the collection: 28 documents for Bliss-alphabet shapes to access Bliss-characters, three documents for Bliss Indicators to access all the Bliss Indicators, one document for different alphabets and one document for Bliss punctuations to access all the Bliss punctuation characters.

The Bliss Logs Collection contains all the event logs of the Bliss Writer system. Everything is logged in the Bliss Writer system. These log data are used to analyse Bliss Writer user studies.